USOFT 磁碟月刊

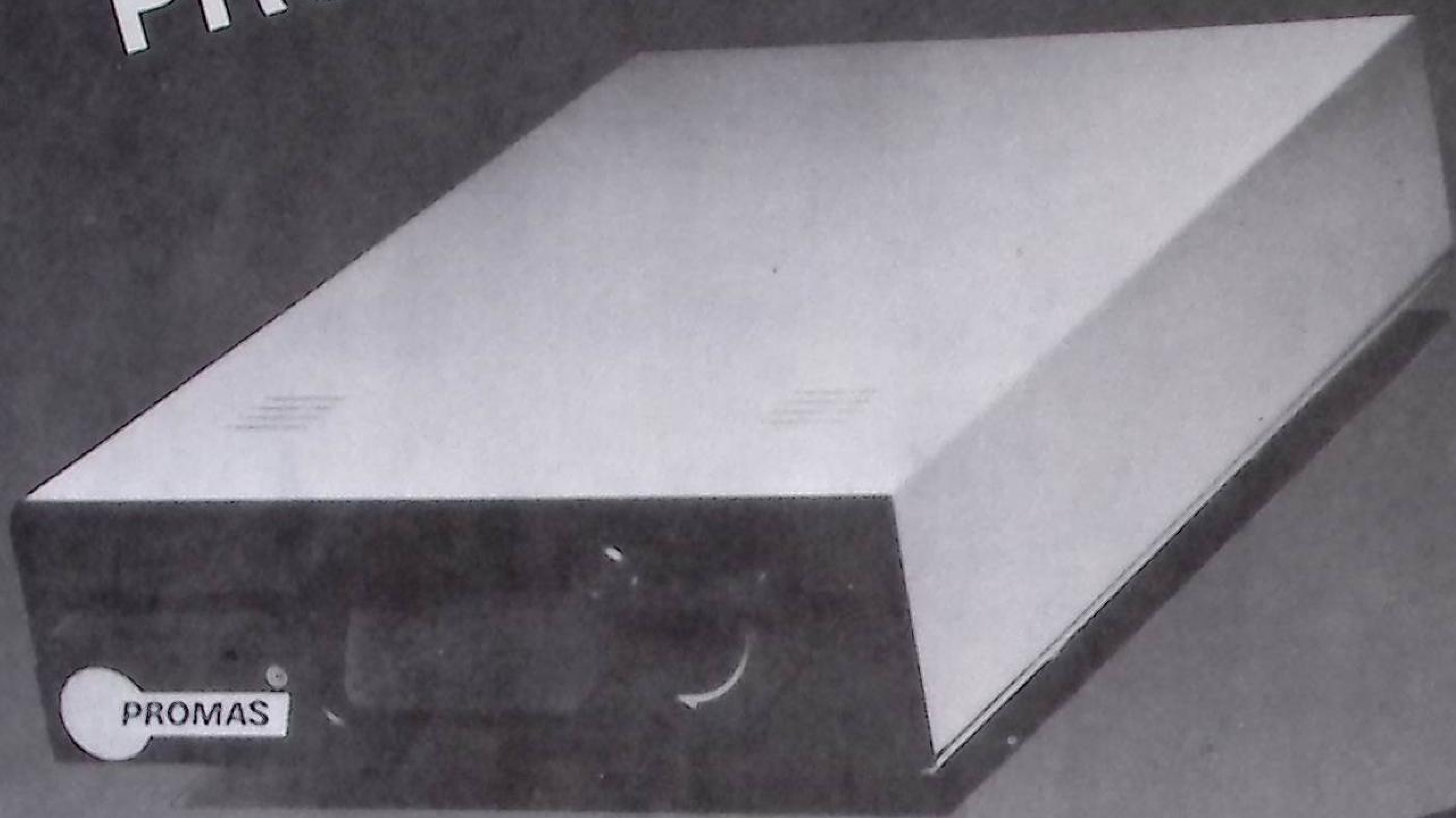
號經已出版

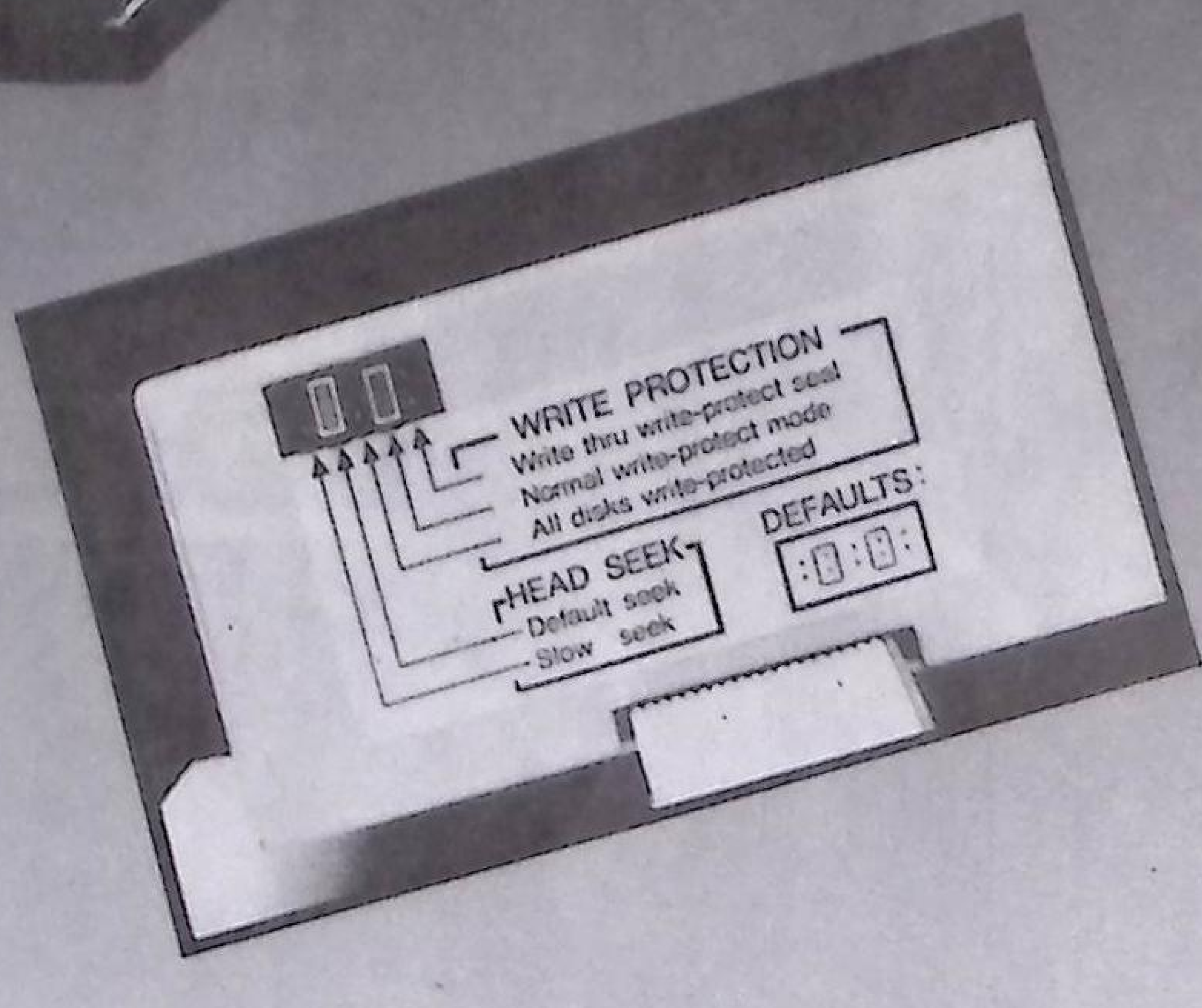
雙面灌錄的磁碟(合共4面)

(11) 無名的遊戲——要讀者自行發掘玩法的遊戲
(12) 最新遊戲魔水晶二號破解方法
(13) 美國最新出版軟件榜
(14) 本港最受歡迎遊戲軟件榜
(15) 供讀者來函的電子郵箱
(16) 商品廣告
(17) 徵求COMPUSOFT 磁碟月刊封面 版頭設計
(18) 一行程式集
(19) CONTROL RESET 使用的方法
(20) 其他…………

新設計磁碟機

PROMAS Model JT





- 獨特"SLOW SEEK"功能，存取資料更準確。
- 內置WRITE PROTECT及HEAD-SEEK速度選擇掣。
- 採用新設計高質素機械部份，可靠程度更高。
- 支架用純不銹鋼製造。
- 完全適用於APPLE II*系列電腦。
- 全日本製造及質素檢定合格。

總代理：
VINCENT COMPUTER CENTRE LTD.
雲生電腦有限公司
九龍尖沙咀柯士甸路22·26號好兆年行6字樓　電話：3-7211288-0　電訊：HX38316 VCC HK

*Trademark of Apple Computer Inc.

只花三元便可改裝 16K RAM 咭 增加狀態顯示器

BABEL

前言

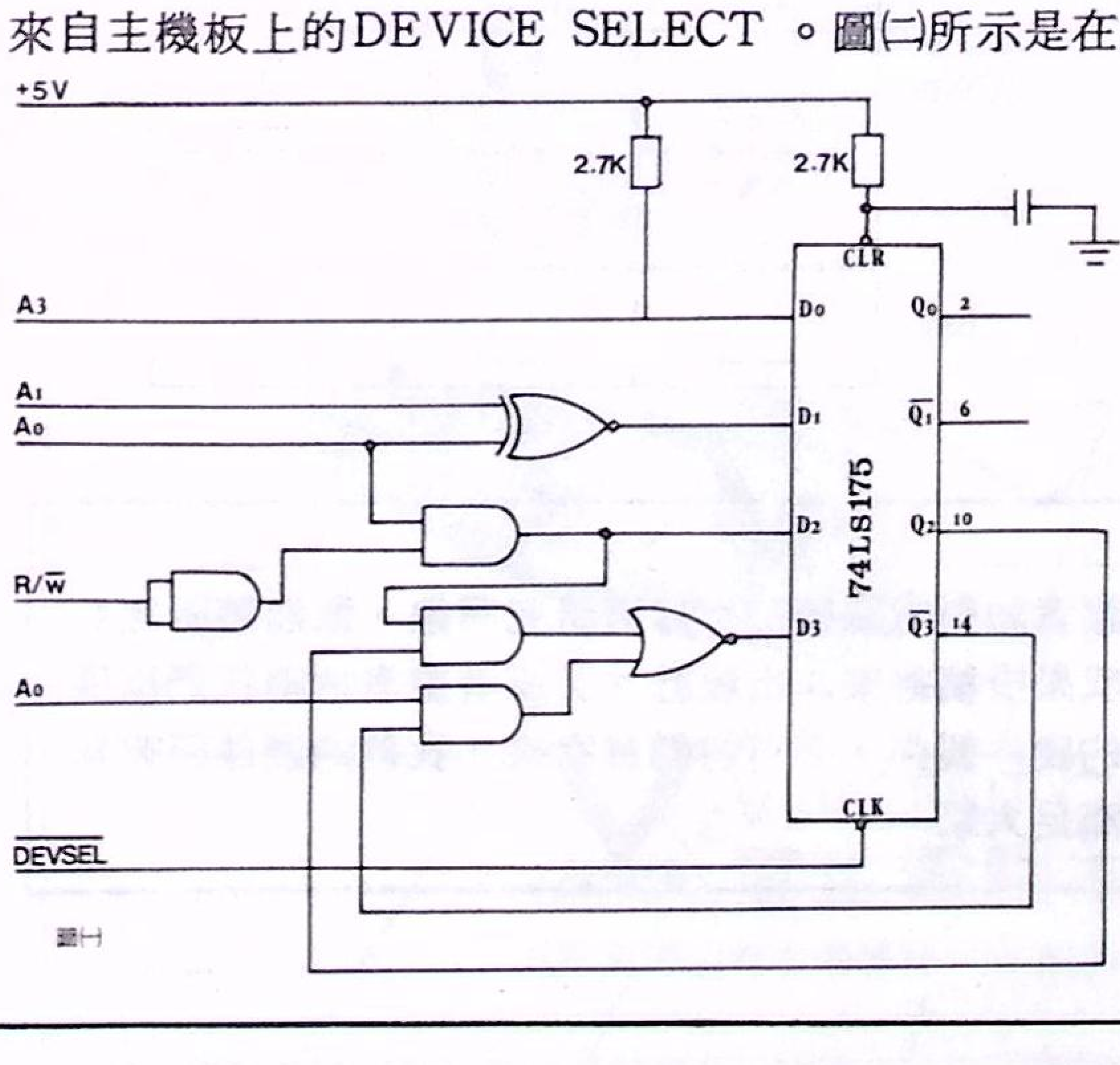
16K RAM CARD 是一塊非常有用的介面咭，相信大部份蘋果用家必定都擁有這塊咭。它既能取代主機板的ＲＯＭ而成另外一個語言系統（例如：PASCAL），又能擴展主機的記憶（例如：CP／M 56K和CP／M 60K，甚至本刊第10期所刊出的RAM DOS程式，也是爲取回主機板上10.5Ｋ記憶體而設），亦可用作其他用途（例如：第四期所介紹的一RAM DISK）。

在本刊第十期中，亦曾經對16K RAM CARD 的控制方法，作一詳細分析，但相信不少對16K RAM CARD 有興趣的朋友，都希望可以從一個直接的方法中，看到＄C080，＄C081 等等指令的效果。市面上有售的RAM CARD，就筆者所見，也有三、四種之多，有APPLE公司的LANGU—AGE CARD，MICROSOFT 公司出產的，也有不用接綫（CABLE）的。其中只有MICROSOFT 的那一種有三粒LED，用以顯示RAM CARD的不同狀態，而用家從三粒ＬＥＤ的狀態，就可以知道究竟RAM CARD 正使用那一個BANK，而咭上的ＲＡＭ是處於READ，抑或是WRITE，從而瞭解到軟件對RAM CARD 狀態的控制了。

但據筆者所知，很多朋友擁有的是不用接綫那一種，因爲不用連接綫，實在很方便。於是筆者便對這塊RAM CAR—D 的綫路進行修改，使到擁有這一種RAM CARD 的朋友，不用再因此而失望，只需花費港幣三元不到的代價，便可以令閣下的RAM CARD 擁有同樣的功能了。

原理概述

幾乎所有16K RAM CARD 中都有一隻74LS175的ＴＴＬ，這隻ＩＣ 就是負責選擇RAM CARD 不同狀態的，它在RAM CARD 中的綫路部份可參閱圖㈠。74ＬS175 內是四個Ｄ型觸發器，分別記着四種狀態的資料，它的時鐘訊號是來自主機板上的DEVICE SELECT。圖㈡所示是在八個不同指令下，RAM CARD 的八種不同狀態。圖㈢所示是在八個不同指令下，74LS175的輸出狀態，而我們正是要利用這些不同的輸出狀態。首先，我們比較圖㈡與圖㈢，便會發現當選擇RAM CARD的BANK 1 的時候，$\overline{Q0}$ 便會輸出〝Lo”，而Qo則輸出〝Hi”。當選擇RAM CARD 的ＲＡＭ是READ狀態時，$\overline{Q1}$便輸出〝Lo”，而Q1則輸出〝Hi”。當選擇RAM CARD 的RAM是WRITE狀態時，Q3 便輸出〝Lo”，而$\overline{Q3}$則輸出〝Hi”。我們便是利用Qo，Q1及Q3這三個輸出來控制三粒ＬＥＤ的亮或滅，使我們可以由此而直接知道RAM CARD 在那一個狀態之下。圖㈣所示，就是我們要附加上RAM CARD 的綫路。



圖㈠

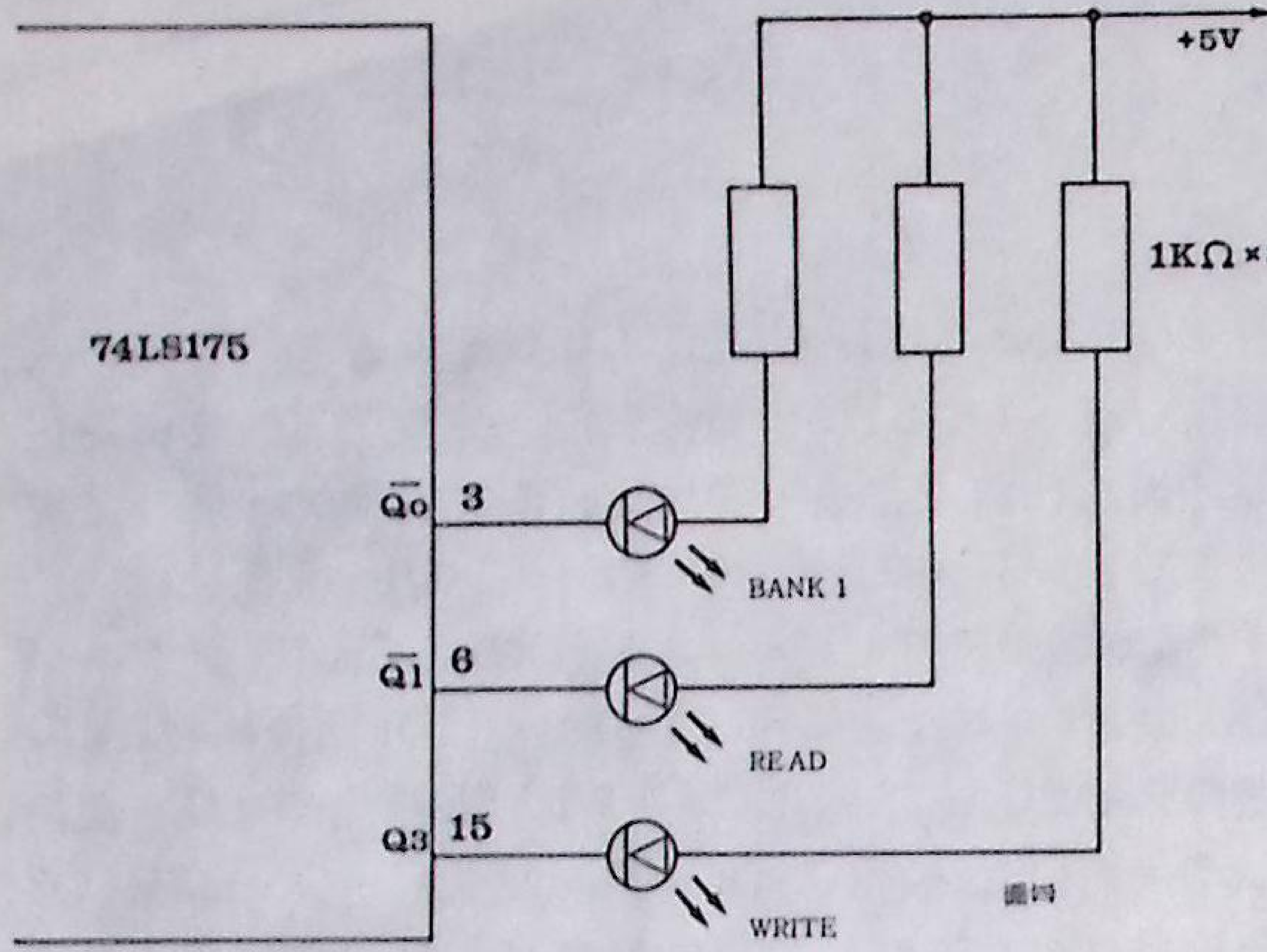
	BANK 1	BANK 2	RAM READ	RAM WRITE
C080		△	△	
C081		△		△
C082		△		
C083		△	△	△
C088	△		△	
C089	△			△
C08A	△			
C08B	△		△	△

圖㈡

▽—H
▲—L

	Q0	$\overline{Q0}$	Q1	$\overline{Q1}$	Q3	$\overline{Q3}$
C080	▲	▽	▽	▲	▽	▲
C081	▲	▽	▲	▽	▲	▽
C082	▲	▽	▲	▽	▽	▲
C083	▲	▽	▽	▲	▲	▽
C088	▽	▲	▽	▲	▽	▲
C089	▽	▲	▲	▽	▲	▽
C08A	▽	▲	▲	▽	▽	▲
C08B	▽	▲	△	▲	▲	▽

圖㈢



圖(四)

裝置方法

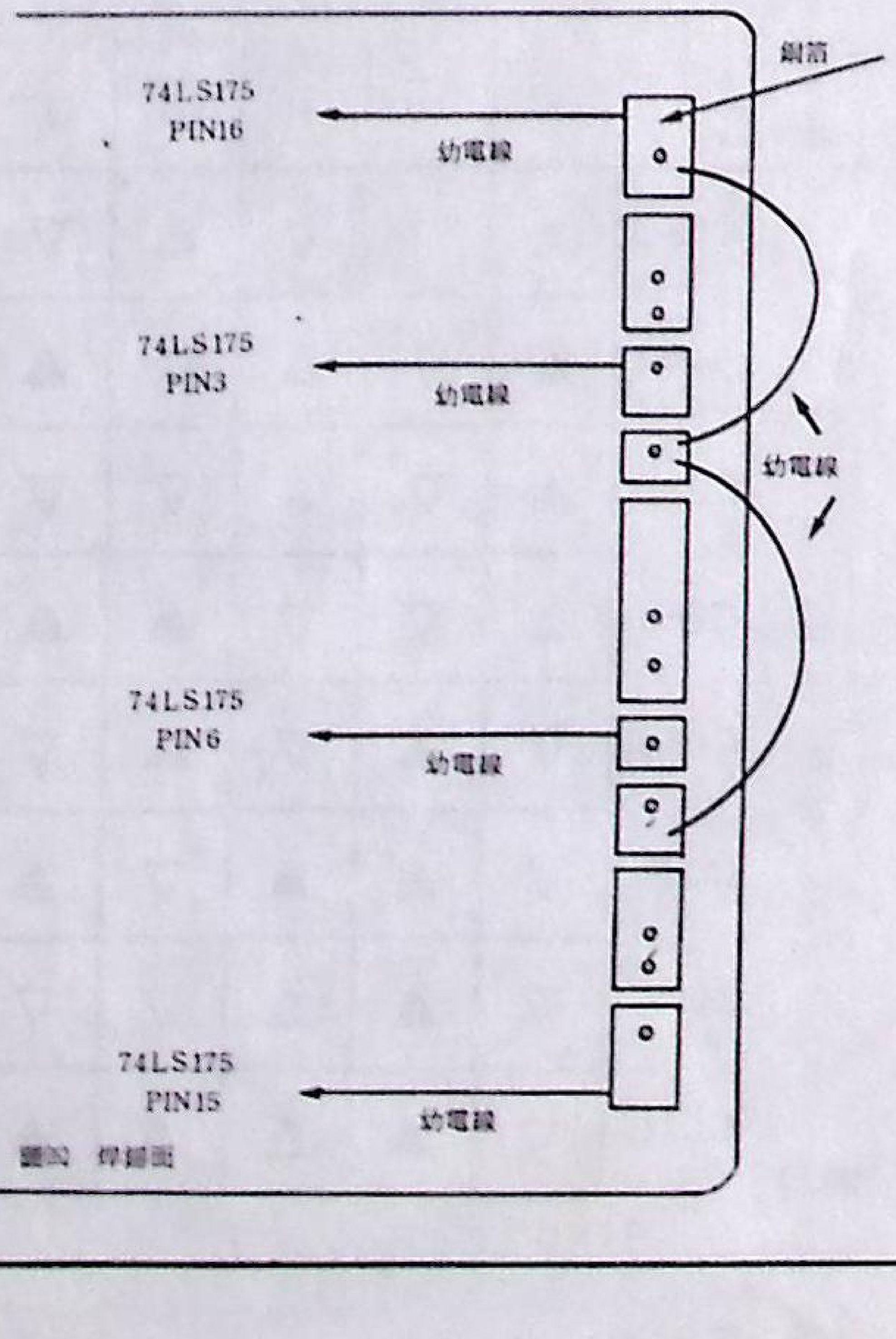
我們需要的零件只是三粒ＬＥＤ，和三支1KΩ 的電阻。這三支電阻是用作ＬＥＤ的限流電阻，考慮到74LS175的負載能力問題，所以便選擇較大阻值，使電流較小，正因爲電流較小，所以各位選購ＬＥＤ時，應選擇較細小的那一種，如果選用大粒的可能會不亮。此外，還需要少量幼電綫作接綫用。

筆者所用的RAM CARD ，在板的焊錫那一面的邊緣，是有一條空置的銅箔，所以我只需輕輕刮去銅箔上的保護油，用刀片介成圖(五)所示一段段，然後鑽上所需的小孔，用以插零件。如果你的RAM CARD 是沒有這一條多餘的銅箔的話，你可以先鑽了零件孔，插上零件後，用幼電綫連接起相關的零件脚亦一樣，零件的排列方法如圖(六)所示。最後，接上所須幼電綫，便可正常工作。

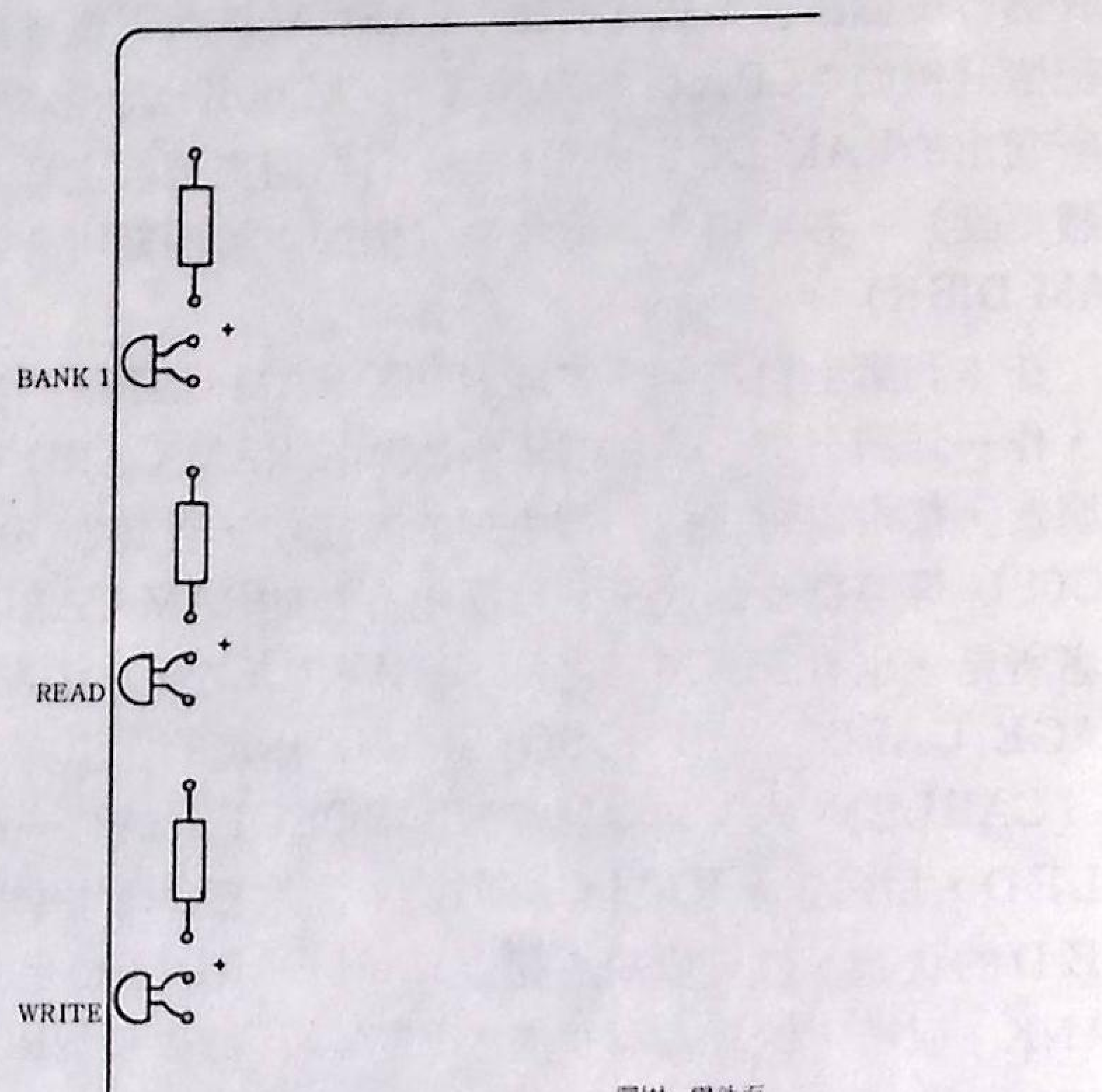
後記

當測試過工作正常之後，便可以用LABEL貼在LED旁以標示出那一粒是指示BANK 1，那一粒是READ，那一粒是WRITE 了。當然，你可以選擇用不同的顏色的ＬＥＤ來代表，則無需貼ＬABEL了。圖(七)所示是工作正常時，不同指令與ＬＥＤ狀態的關係。

最後，各位可有留意到，原來開機之後，RAM CARD是處於BANK 2 的WRITE RAM 狀態，即是＄C081 之後的狀態；而又有時＄C 081 是要鍵入兩次以上才有效的呢？有時間便思考一下吧！■



圖(五) 焊錫面



圖(六) 零件面

	BANK 1	RAM READ	RAM WRITE
C080		△	
C081			△
C082			
C083		△	△
C088	△	△	
C089	△		△
C08A	△		
C08B	△	△	△

圖(七)

各位讀者如對電腦硬件有特別研究興趣，歡迎將研究心得或改良成果投稿寄來本出版社。又或者讀者希望我們提供什麽類形的硬件製作，亦可將題目寄來，我們的硬件研究組定會盡力滿足大家。

使你的 APPLE DOS 更加多彩多姿

何鴻威

大家不知是否發現在DOS 裏面尙有兩塊記憶體未使用到，一爲（$BA69—$BA95），二爲（$BCDF—$BCFF)，所以筆者馬上就有個靈感，如果我們能使APPLE 在開機執行BOOT DOS的過程中，弄個中文名到螢幕上，然後使字形由小變大，同時再加上音响效果，這樣一來我們就可在磁碟上擁有自己專門的〝商標〞，而且是個漂亮的彩色圖案。或許各位讀者認爲自己可以用BASIC 寫個程式，然後將之存入HELLO程式中，因爲在開機後，APPLE 會自動RUN HELLO程式，如此一來，也可以達到上述同樣的效果。但各位不要忘記，當別人從你那兒 COPY 程式的時候，他可以非常輕易的去掉或更改你的程式，這樣你的目的也就無法達到了。倒不如讓筆者教大家這一招絕招，就是將程式混入DOS 中加以保護，因爲DOS 本身是非常複雜不易了解的程式，若沒有相當的程度而想去改DOS 的話，說不定整片磁碟就會死在那裏，而無法動彈。

首先談談這個程式的組成，我將之分爲兩部份，第一部份爲（$BCDF—$BCFF) ，是主程式所使用的，第二部份爲（$BA69—$BA95)，是存放圖形資料。因爲主程式只有33個BYTE 的空間，所以我盡量找出MONITER 與APPLESOFT 內部的副程式來使用，以求用最少的記憶空間來達成最多的功能，而且爲了讓各位讀者在看主程式時，有一目了然的感覺，我先說明一下，各個副程式的功能及啓始地址，還有事前應準備的工作：

(1) $F3D8 ：從此地址開始執行的副程式，會自動設定爲高解像度的第二頁繪圖螢幕，然後清除畫面爲黑色，如同BASIC 的HGR2指令。

(2) $F411 ：此副程式會將高解像度的繪圖游標指到你所規定的座標地址上，但是不執行繪圖動作，只等待進一步的指令而工作。
水平座標存入Y，X暫存器中
垂直座標存入A暫存器中

(3) $F601 ：此副程式功能與BASIC中DRAW指令一樣，其中心點座標須先由$F411 副程式確定好，而造型表的啓始座標由Y，X暫存器所指定，將旋轉角度（ROT）存入A暫存器中，放大向量（SCALE） 存入$E7 地址中。

(4) $FF3A：此副程式會使 APPLE 發出一聲〝嗶〞的聲音。

各位看到這裏，也應該心裏有數了，起碼應該知道五、六成，若尙還摸不到頭緒的話，也不用著急，其實下面的主程式的說明，只要懂得BASIC 與了解一點ASSEMBLY的觀念，就可以摸透整個程式的來龍去脈，若有未學過ASSE—MBLY LANGUAGE 的讀者，也不用氣餒，你只要將主程式的MACHINE CODE 小心打入，然後自己設計圖形就可以了。

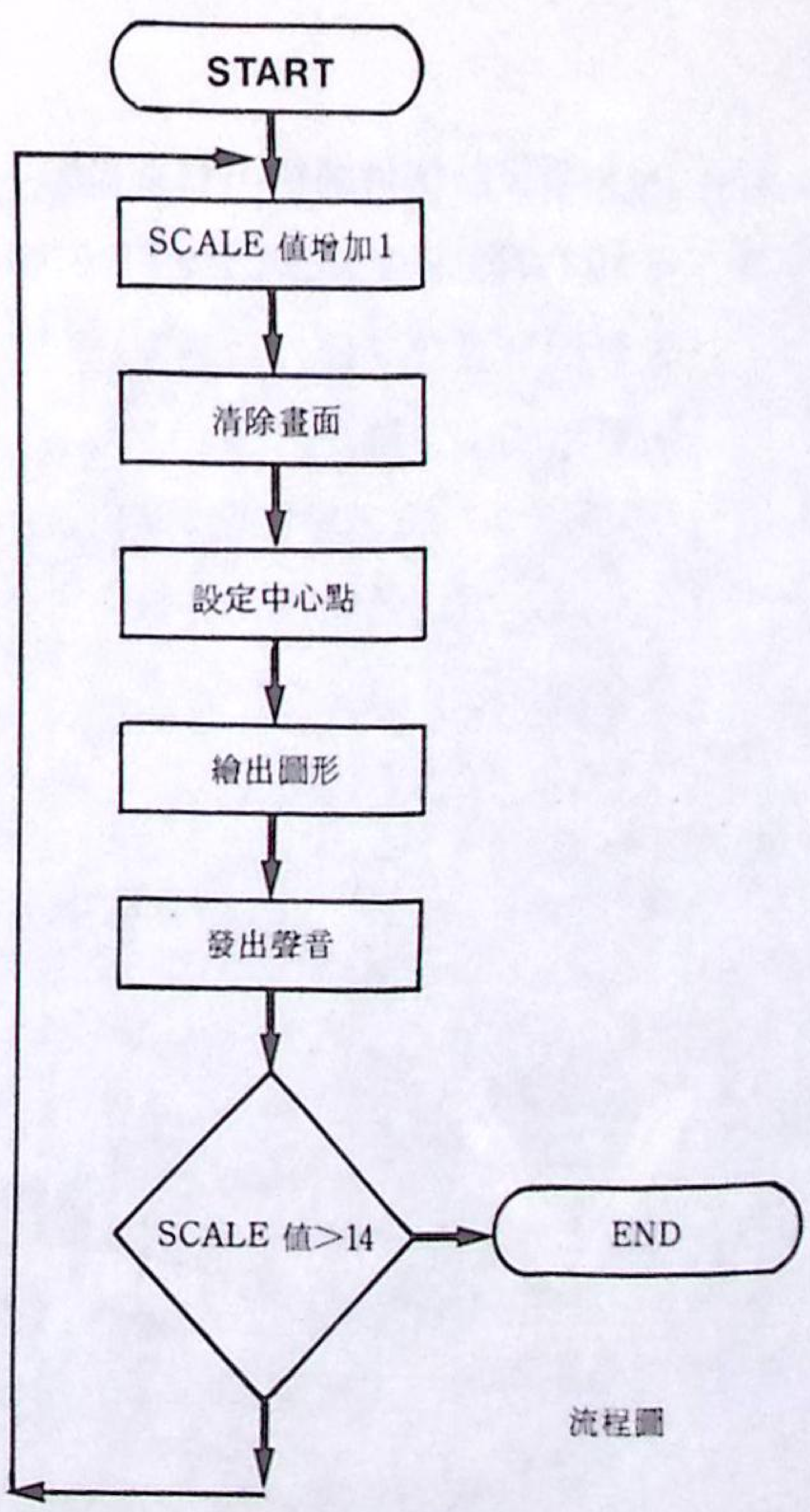
各位在看主程式時，要注意程式裏面有一個（JSR $B78D）的指令，它只是用來擴充記憶體，並無其他用途，而其範圍爲（$B78D—$B792)。另外程式本身省略許多非必要性的指令，這全因記憶體實在有限的原故。

程式首先將X暫存器的值放入$E7中，代表SCALE的大小，然後去執行HGR2（$F3D8)的動作，再由$F411副程式定出中心點位置，將ROT的值存入A暫存器，由

主程式

```
*BCDFL

BCDF-   E8          INX
BCE0-   E8          INX
BCE1-   86 E7       STX   $E7
BCE3-   20 D8 F3    JSR   $F3D8
BCE6-   A9 65       LDA   #$65
BCE8-   A0 00       LDY   #$00
BCEA-   AA          TAX
BCEB-   20 11 F4    JSR   $F411
BCEE-   A9 00       LDA   #$00
BCF0-   A0 BA       LDY   #$BA
BCF2-   A2 69       LDX   #$69
BCF4-   20 8D B7    JSR   $B78D
BCF7-   A6 E7       LDX   $E7
BCF9-   E0 0E       CPX   #$0E
BCFB-   30 E3       BMI   $BCE0
BCFD-   4C D4 A7    JMP   $A7D4
```



流程圖


PROMAS 超薄磁碟機
Model JA
PROMAS
・全新設計機械部份，特別柔順。
・40 TRACKS, 143K BYTES 儲存量。
・低噪音設計，操作寧靜。
・體積細小，機身輕巧。
・Apple II* 及相容電腦專用。
・全日本製造及裝嵌，品質檢定合格。
VINCENT COMPUTER CENTRE LTD.
雲生電腦有限公司
九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK
* Apple II is a registered trademark of Apple Computer Inc.

使你的APPLE DOS更多彩多姿

X，Y暫存器確定造型表的地址爲＄BA69，等到上述準備完畢後，才去執行DRAW（＄F601）的動作，然後"嗶"一聲，最後比較＄E7的値是否大於14，如果小於則重來一次，否則就結束程式。

接下來介紹第二部份（＄BA69—＄BA95）圖形表示法，其與造型表（SHAPE TABLE）略有不同，原本造型表的構造爲前面一段存放索引檔，其中包括造型的數目，相對距離等。但是在這裏存放在DOS區域（＄BA69—＄BA95）的造型表，則必須省略前一段，而直接存入圖案的繪圖向量。若是各位讀者對於做造型表不甚了解時，應重新看一下APPLE的手册，以便充份利用每一個空間，來把美麗的圖案完整的建立起來。我以一個「張」字爲例，以下是其造型表：

〔張〕字圖形表

```
*BA69.BA95

BA69-  2D 2D 2D D8 38 BF 26
BA70-  24 2C 6D C0 3D 3F 37 E6
BA78-  C3 DB 2D 35 36 3F 37 36
BA80-  2D 35 36 06 3E 3F 44 41
BA88-  49 EC B4 36 2E 49 C5 DF
BA90-  45 C7 4D 47 05 00
```

現在我們討論一下，要如何使APPLE於開機的時候就執行我們所建立的程式。因於開機前，APPLE記憶體並無任何程式，以其本身以BOOT的方法，以獲得DOS程式，當所有BOOT的過程結束後，會首先跳入冷啓動（COLD START）位置而去執行，其位置爲（＄9D84），我們可以在其下方的地址選擇適當的地點跳出，去執行我們的程式，再跳回來，這樣就可以了。以下是其原理：

原來的 9E0C—20 D4 A7

更改爲 9E0C—20 DF BC

這也是爲何主程式最後一個指令爲JMP ＄AD74的原因了。

看到了這裏，各位大可鬆一口氣了，如今所面臨的最後一個問題是要如何將已經動過手脚的DOS存回原磁碟上，或許各位認爲用INIT指令就可解决這一難題。不錯，但那只能用於新的磁碟片上，却無法用於已經擁有程式的磁片上。關於這一點，我是利用〔COPYⅡ PLUS 4.3〕這片磁碟的帮助來解決的，其操作手册市面上有賣，各位讀者對於使用方面應不成問題，但我在這裏提供一些附加資料，以免空有磁碟却無法使用。

（＄9E0C—＄9E0E）⇒第0磁軌，第13磁區

（＄B78D—＄B792）⇒第1磁軌，第06磁區

（＄BA69—＄BA95）⇒第0磁軌，第04磁區

（＄BCDF—＄BCFF）⇒第0磁軌，第06磁區 ■

誤購水貨 得不償失

所謂"水貨"，就是一般投機商人，直接由原產地而不經本地代理商進口的貨品。通常會比正常之行貨售價略低，主要是因爲水貨售價不需加上日後之保養，廣告及存貨等成本，而且水貨每次進口量少而銷量快，加速現金吐回而免却利息之計算。所以，以略低之價錢出售，當然吸引不少之消費者。

價錢低，並不是這樣理想的。所謂一分錢一分貨。有很多表面察覺不到的問題往往存在於水貨之中，這就是水貨與行貨之分别了。

一般代理商獲廠方委任代理時，大部份均有廠方的專業維修訓練，或是提供"維修手册"，零件也供應完備。而有些零件更是廠方特别製造，而在市面上不會發售，只供給代理商作維修之用。通常代理商在免費保用期後，都以低廉而合理價錢，向用戶收取所更換之零件費，令顧客滿意。日後更會提供顧客最新產品資料，但"水貨"便沒有此種服務。

通常代理商是拒絕修理"水貨"或經他人修理不成功之貨品，倘若代理商願意替"水貨修理"，也祇好將所有零件更換，或甚至機壳之外，整個更換。這樣做法，修理費用必定昂貴。仔細計算下，倒不如買另一部新貨品還化算。

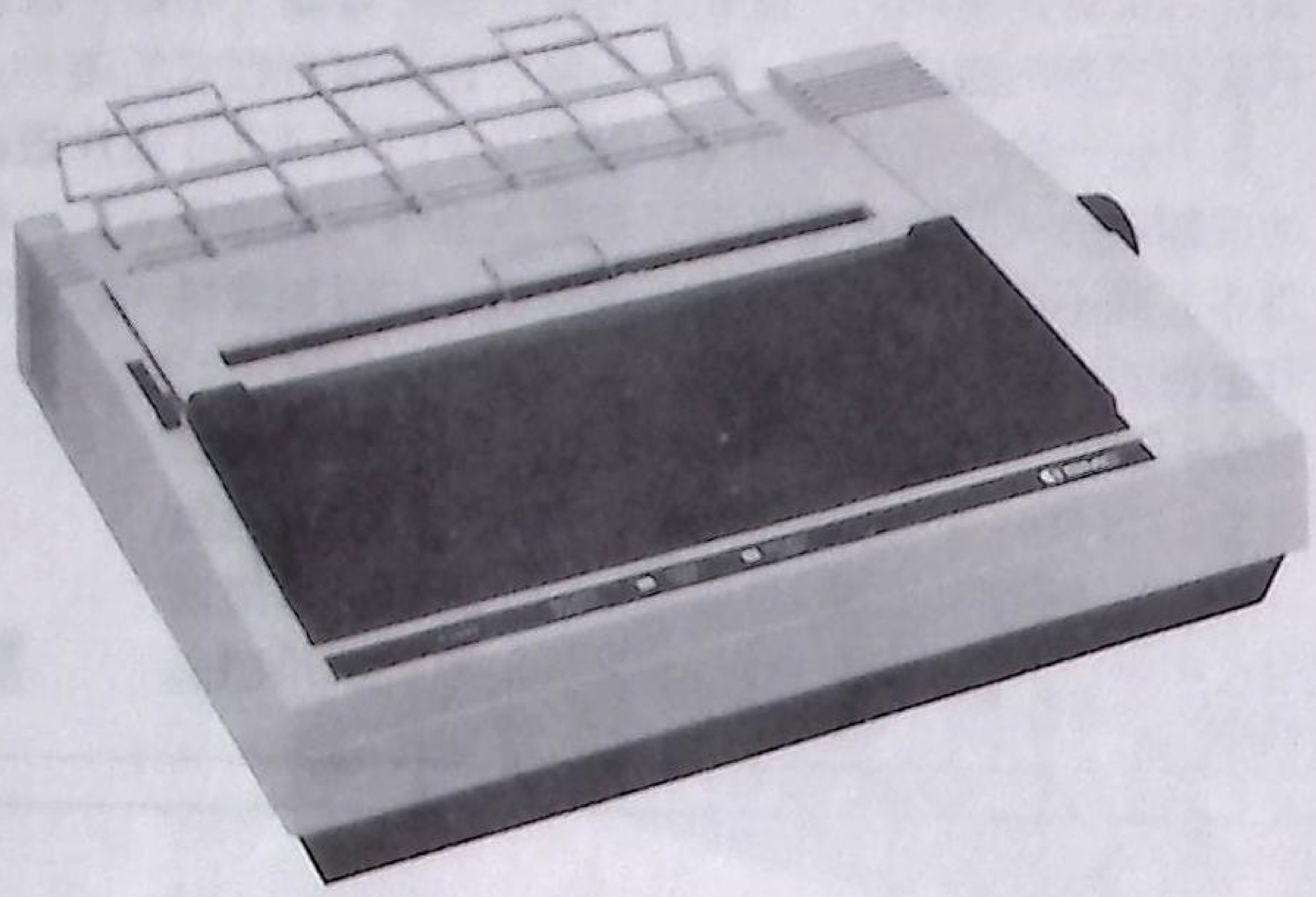
零售商方面也請注意。如顧客日後發覺所購得之商品爲"水貨"時，不但對貴號所售的貨品失去信心，甚至會因此而失去該顧客。又"水貨"如大量充塞市場，代理商必會考慮降價來打擊"水貨"銷量。到頭來，"行貨"售價可能與"水貨"相同或甚至略低。如零售商購入大批"水貨"後發覺"行貨"售價調低時，所存之"水貨"便很難售出，那時眞正得不償失。

怎樣才可分辨"行貨"及"水貨"

1. 經常留意商品廣告上之代理商名號，一遇疑問，立即向該公司查詢。因非代理商之商號是不能隨便冠上總代理或經銷商的名稱，否則會招至法律上控訴之問題。
2. 留意型號，一般廠方爲維護該地之代理商，均將出口及內銷之貨品冠以不同型號，以資識别。
3. 留意電源供應之電壓說明是否與該地所用之電壓相同。如有改變及改換，均大多不是行貨。
4. 有些商品之包裝甚至印明代理商號名稱。
5. 最重要一點是要索取代理商之全新保用證，而不是原廠或零售商之保用證。因爲在原產地發售之貨品，大部份是原廠直接保用，但這些保用證不包括外地用家。

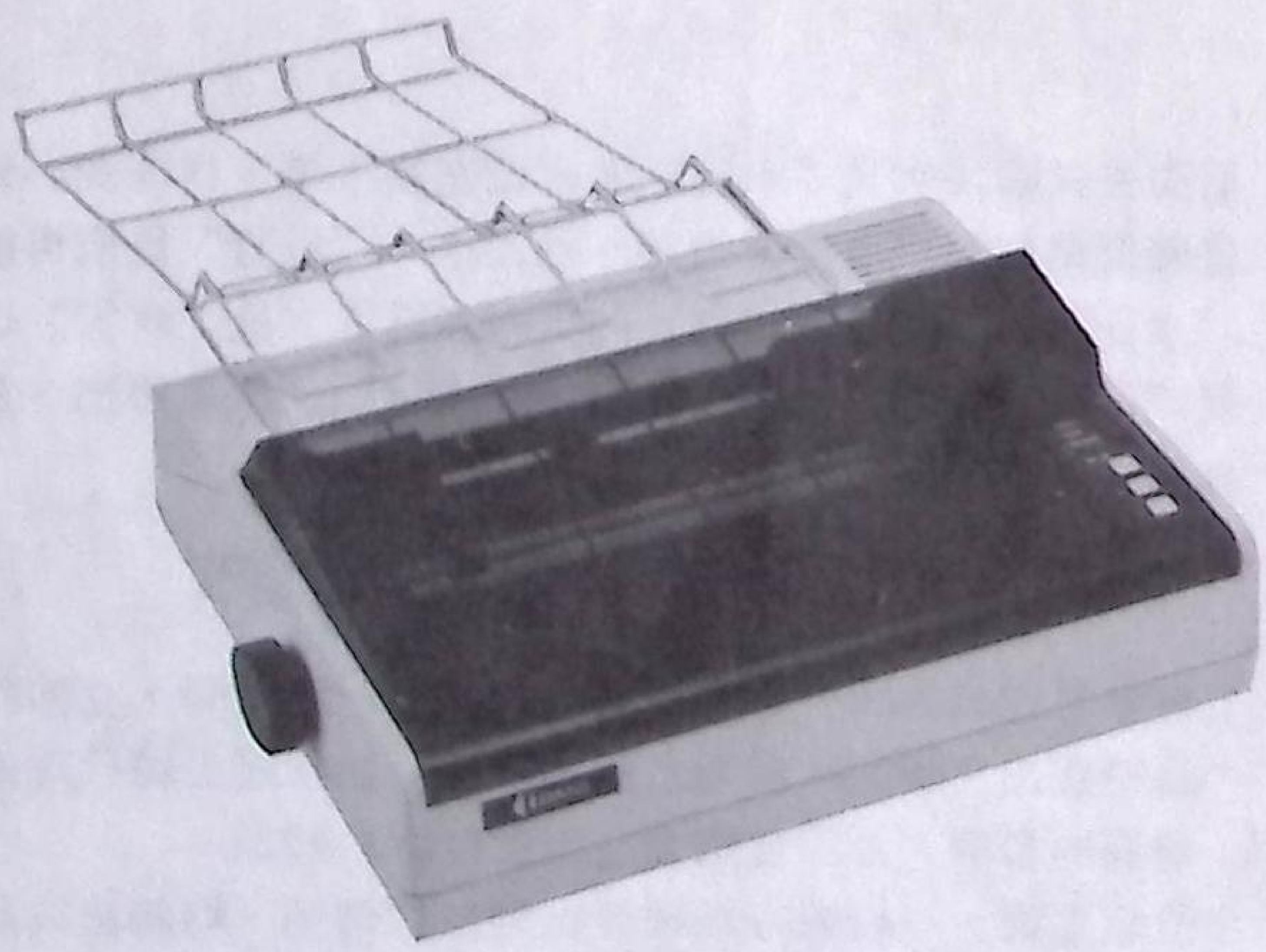
最後，希望用戶在購貨前小心考慮一下，免招損失。請記著：「代理商專業保用，才是最可靠之保證。」■

BMC
超值印字機



BMC BX-100點矩陣式印字機

- 高速來回式印字，每秒可印100字。
- 可印四國文字。
- 多種獨特印字功能，別於同類型產品。
- 盒式墨帶，壽命特長，可連續印達2,500,000字。
- 內置CENTRONICS平行式介面接口，可另加配RS 232C串行式介面接口，適合任何電腦。
- 全日本製造。



BMC BX-80 點矩陣式印字機

- 印字速度，每秒80字。
- 可用普通紙及電腦紙。
- 兩種字體，每行可印71-142個大小文字。
- 可編印中文字及各種圖案。
- 標準CENTRONICS平行式介面接口，適用於一般電腦。
- 全日本製造。

BMC BX-1000 點矩陣式印字機

- 高速來回式印字，每秒可印100字。
- 功能與BX-80相同。

總代理：

VINCENT COMPUTER CENTRE LTD.

雲生電腦有限公司

九龍尖沙咀柯士甸路22-26號好兆年行608室　電話：3-7211288-0　電訊：HX38316 VCC HK

D3.3 O S 專欄 甘貴祥

40磁軌抄錄程式

(適合單磁碟機用戶)

莫信

編者註：DOS 3.3 專欄近幾期都是講述40磁軌的問題，日前我們收到讀者莫信君的一篇投稿介紹一個40磁軌抄錄程式（EVERY TEN COPIER），程式對於單磁碟機用戶相當適合，而且作者在文字說明解釋上相當的詳盡而清楚。我們特別與DOS 3.3專欄欄主甘貴祥君商量，決定將這篇文章在這個專欄上發表。

40磁軌的FORMAT 格式將會越來越流行，無他，因爲可以增加磁碟的儲存量，而且40 TRACK磁碟機越來越普遍，相信讀者都同意這點吧！

此程式是專爲一些只有48K 電腦和一部磁碟機的用家而設計的，它的好處是每次能夠抄錄10條磁軌，以40軌的磁片，祇要 4 次抄錄便成了，無可否認，現時通行的抄錄程式很多，例如：NIBBLE AWAY，DISK MUNCHER，COPY][PLUS，LOCKSMITH 等，但它們的運行和操作原理，各位可知道清楚嗎？現在就讓我們藉着這個EVERY TEN COPIER，對它們進行剖釋，抄錄程式的抄錄方法很多，而最普遍的便是利用ＲＷＴＳ副程式（READ OR WRITE A TRACK AND SECTOR），直接利用機器語言去處理 磁碟的讀寫工作。在下列的解釋中，任何放在＄之後的數字均表示十六進位數字。

這個〝READ OR WRITE A TRACK AND SECTOR〞副程式，允許使用者經由機器語言，從磁碟片上的任一磁軌或磁區上讀取訊息或將訊息寫上去。爲了使用這個ＲＷＴＳ副程式，使用者必須首先建立一個IOB（輸出／輸入控制區）表，及一個〝設備特性表〞。這個IOB表會告訴ＲＷＴＳ副程式，那一個插座及磁碟機號碼放置了磁碟機，磁碟片的磁册號碼爲何，要處那一個磁軌及磁區，到底是要讀取或寫資料到這一磁碟片上，而設備特性表則提供一些有關操作磁碟機所需要的訊息給ＲＷＴＳ副程式。

除了IOB 表和設備特性表外，我們必須寫一個〝控制副程式〞，並且存到記憶體中，這個副程式必須 JSR到ＲＷＴＳ的起始位址去，在DOS MANUAL中，我們可找到ＲＷＴＳ的起始位置是在＄3D9，但位址＄3D9並不是ＲＷＴＳ眞正的進入點，利用監督程式指令3D9L 看看位址＄3D9 放著什麼？原來是4C B5 B7（卽JUMP＄B7B5），眞正的進入點是＄B7B5，所以使用ＲＷＴＳ副程式時，祇要JSR＄B7B5 便成了，在使用ＲＷＴＳ副程式前，Ａ及Ｙ暫存器必須包含ＩＯＢ表的起始位置，而Ａ暫存器必須包含著高位址ＢＹＴＥ，Ｙ暫存器則包含低位址ＢＹＴＥ。

說了這麽一大堆的文字，到底怎樣使用ＲＷＴＳ副程式呢？待我們用EVERY TEN COPIER來舉例，程式是由位址＄800 開始的，而〝控制副程式〞是由＄A00開始存放，看看它是一個怎樣的一個樣子：

```
0A00-  A9 0A     LDA #$0A    A暫存器存入$0A
0A02-  A0 11     LDY #$11    Y暫存器存入$11
0A04-  20 B5 B7  JSR $B7B5   呼叫RWTS副程式
0A07-  60        RTS
0A08-  60        RTS
0A09-  00        BRK
```

它將A暫存器存入＄A，Ｙ暫存器存入＄11，亦卽是說它設定ＩＯＢ表的開始位置在＄Ａ11，然後跳到ＲＷＴＳ副程式去，再看看ＩＯＢ表（圖一），這個ＩＯＢ係用來處理插座編號 6，及 1 號磁碟機，以便將第11號磁軌的 0 號磁區資料讀到記憶體從位置＄9D00 開始的256 個ＢＹＴＥ上。（留意：位址＄9D00 是被ＤＯＳ佔用的，所以將資料讀到＄9D00 時便會產生錯誤，各位若自己寫程式時，應將＄Ａ1Ａ改作小於＄96的代碼，而爲何EVERY TEN COPIER將緩衝區開始位址設定爲＄9D00 ，下文會有交待。）

而設備特性表亦須具備，我們將其放在ＩＯＢ之後的位置＄Ａ22上。在ＩＯＢ表中位置＄Ａ17及＄Ａ18係指示到設備特性表的起始位址。請參看圖二，若各位對ＩＯＢ表及設備特性表作進一步的了解，請參看圖三和圖四的ＩＯＢ格式表和設備特性格式表。

說完ＲＷＴＳ的用法後，待我們將主程式逐點分析，看看主程式流程圖（圖五），開始時我們CALL 2548（＄9F4）進入程式，隨之便是破壞ＤＯＳ系統。破壞ＤＯＳ系統，攪笑，這不是攪笑，若我們只抄錄 7 條磁軌或 8 條磁軌，我們便不需要這樣做，但每次抄錄10條磁軌，需要記憶體極多，0頁、1頁和第 2 頁都不能用，而位址＄400 至＄7FF是作顯示用的，位址＄800 至＄CFF是放有EVERY TEN COPIER 程式，因此資料緩衝區便由位址＄Ｄ00開始，10條磁軌共包含256×16×10 BYTES，故此資料緩衝區的長度是由位址＄D00至＄ACFF，而位址＄9D00至＄BFFF是存放ＤＯＳ系統的位置，亦卽是說資料緩衝區會蓋掩ＤＯＳ系統其中的一部份，所以破壞ＤＯＳ系統是必需的，否則便不能由磁碟讀資料到電腦記憶體的位址＄9D00至＄ACFF去，那ＲＷＴＳ副程式呢？它是ＤＯＳ系統的一部份，DOS系統玩完了，豈不是ＲＷＴＳ副程式也玩完？不然，資料緩衝區祇是至位址＄ACFF，而ＲＷＴＳ副程式是放在＄Ｂ7Ｂ5之後，故此對ＲＷＴＳ副程式的使用是沒有影響的。

當ＤＯＳ系統受到破壞，電腦會將POINTER指向零頁

40磁軌抄錄程式（適合單磁碟機用戶）

圖一：ＩＯＢ表

位　置	代碼	用途
＄Ａ11	01	IOB形態的指示器，必須爲＄01
＄Ａ12	60	插孔編號乘上16
＄Ａ13	01	磁碟機號碼
＄Ａ14	00	預期的編冊數目
＄Ａ15	11	磁軌號碼
＄Ａ16	00	磁區號碼
＄Ａ17	22	設備特性表的低順序位元
＄Ａ18	0A	設備特性表的高順序位元
＄Ａ19	00	資料緩衝區的開始位址的低順序位元
*＄Ａ1Ａ	9D	資料緩衝區的開始位址的高順序位元
＄Ａ1Ｂ	00	沒有使用
＄Ａ1Ｃ	00	沒有使用
＄Ａ1Ｄ	01	命令編碼，＄01＝ＲＥＡＤ
＄Ａ1Ｅ	00	錯誤編冊數目
＄Ａ1Ｆ	00	實際的編冊數目
＄Ａ20	60	先前被處理的插孔號碼
＄Ａ21	01	先前被處理的磁碟機號碼

圖二：設備特性表

位　置	編碼	
＄Ａ22	00	設備形態編號（放一＄00）
＄Ａ23	01	每一磁軌的現象數（此放一＄01）
＄Ａ24	EF	計時（此放一＄EF）
＄Ａ25	D8	計時（此放一＄D8）

圖三：設備特性表的格式

BYTE號碼	名　　稱	用途
1	ＤＥＶＴＰＣ	設備種類編號，告訴這是何種的設備，當使用ＤＩＳＫⅡ時＄00就須存在這個ＢＹＴＥ上。
2	ＰＰＴＣ	在此ＢＹＴＥ上必須存放着＄01
3＆4	ＭＯＮＴＣ	計時運轉的補充，以100微秒爲區間，當使用ＤＩＳＫⅡ時，在ＢＹＴＥ３內存著＄ＥＦ，在ＢＹＴＥ４內存放著＄Ｄ8

的位址＄８（每種ＤＯＳ版本都會使POINTER指向不同的位置），記得我們開頭時要CALL 2548 嗎？看看位址＄9F4（2548）存放著什麼？它將＄4C、＄80、＄08 存在位址＄８、＄９、＄Ａ，使POINTER指向位址＄８時，作出JUMP＄880 的反應，若各位的版本不同，POINTER指向不同的位置，我們便需要改變＄4C、＄80和＄08的存放位置了，如何查出POINTER指向位置，衹要將０頁的記憶體全部設爲０，然後執行JSR＄A00 ，這時ＤＯＳ系統受到破壞後，POINTER 便會指向零頁的指定位置，但由位址＄0至＄ＦＦ都被設爲０，所以程式被停住了，並顯示停住的位址，若被停住的位址顯示爲＄10，那麽POINTER應是指著＄Ｅ（＄10－＄２），所以我們衹要將＄4C、＄80、＄08存在位址＄Ｅ、＄Ｆ和＄10便可以如常執行了。進入位址＄880 後，有兩個指令，一個是JSR＄A00，另一個是JUMP ＄800，JSR＄A00 是要再一次的對ＤＯＳ系統作破壞，JUMP ＄800 則作圖像顯示，完成圖像顯示後JUMP＄B03，重新設定HIMEN 值和磁軌、磁區編號，命令代碼，資料緩衝區開始位置等，然後執行＄B24等的指令。

由位址＄B24開始，我們進入了程式的廻路，而位址＄B30的JSR＄A30 指令，便是呼叫讀入指示和寫出指示副程式，將〝INSERT DESTINATION DISK〞和〝INSERT SOURCE DISK〞等字句顯示到螢幕上，指示我們應作的步驟，隨後由位址＄B33至＄B51，它設定ＣＨ值爲０，ＣＶ值爲４，JSR＄FC22 副程式（這個副程式名字是VTABZ，用處是運算ＣＨ及ＣＶ值到相對的VIDEO RAM的位址），JSR＄FD35 才是取讀鍵盤副程式，當我們按鍵時，鍵盤脈冲便存到位址＄9FF，我們將它與SPACE BAR和字元０的ASCII 碼比較（SPACE BAR的ASCII 代碼是＄Ａ０而字元０是＄Ｄ１），相對作出反應，若代碼是＄Ａ０便進入工作區（JUMP＄B5F）。

工作區的主要任務是讀入和寫出資料，其次便是作寫出顯示、讀入顯示、完成顯示和測試工作，讀入和寫出資料是由位址＄Ｂ79的JSR＄A00完成，而顯示的工作是由JSR＄AB0完成，JSR＄AB0 之前，我們需要將字元碼放入位址＄9FE，＄D7、＄D2、＄AE 便是字元〝Ｗ〞、〝Ｒ〞和〝　〞的代碼。測試SECTOR編號和TRACK編號是由位址＄B8A至＄BA5完成，若TRACK 編號等於TEST（TEST是一個變數，是用來控制每次的抄錄剛好10磁軌，所以開始時TEST是放著＄Ａ）則驗查ORDER是１或２。

若是１，則設ORDER碼爲２，開始磁軌爲T'，開始資料緩衝區爲＄D00，最後跳至＄B24進行WRITE的動作。

若是２，則設ORDER爲１，開始磁軌不變，T' 等於TEST，開始資料緩衝區爲＄D00，增加TEST爲TEST＋＄Ａ，準備進行第２次讀寫工作，同時，我們需要測試是否將所有TRACKS 的讀寫工作完成，若是，則跳至全部完成顯示；若不，則進入＄B24。

看過流程圖後，細心的朋友都會發覺這個EVERY TEN COPIER是爲40 TRACKS的磁碟的抄錄而設計的，其他磁軌數目的磁碟是不適用的，而且完成一次讀寫工作後，不能立刻作第二次的抄寫，但它是可以補救的，各位可看到流程

40磁軌抄錄程式（適合單磁碟機用戶）

圖有4個標記，第一個標記與第2個標記是一對的，而第3個和第4個標記是一對的，祇要我們在①加上磁軌總數的設定而②加上對T＝磁軌總數的測試，那麼這個程式便可以爲一些只有35磁軌或其他磁軌的磁片作抄錄。而在③和④加上選擇是否重新再來一次的指令，便立以第2次、第3次的抄寫而不需要再重新BLOAD EVERY TEN COPIER和CALL 2548，試試改動它成爲自己適合的程式罷。若各位有2部磁碟機，怎樣去改動它，想想罷？

最後，我要提醒各位，被抄錄的磁碟是要被INIT過的，並且要與SOURCE DISK 的格式一樣，亦即是說，若SOURCE DISK是40 TRACKS，那麼被抄錄的磁碟也是要40 TRACKS，否則這個抄錄並不會成功，切記。還有，若你們自己打PROGRAM 入碟，一定要非常小心，否則或會破壞你自己的磁碟，例如將ORDER 指令寫錯爲2或其他指令碼。

圖四：IOB的格式（FORMAT OF IOB）

BYTE數	名稱	用途
1	IBTYPE	告訴RWTS副程式；這是那一形態的IOB，必須是$01。任何其它形態的代碼，可用來定義。
2	IBSLOT	必須包含插座號碼乘上16的數值，這插座就是磁碟機控制卡的放置位置。例如使用5號插座，則此位置必須存放著$50。
3	IBDRVN	必須包含被處理的磁碟機的編號——爲$01或$02。
4	IBVOL	被處理的磁碟片之磁册號碼必須存放於此。磁册$00將自動被設定給任何的磁碟片。
5	IBTRK	被處理的磁軌號碼（正常爲0到34）被存放於此。必須在$00到$22之間（若磁碟片只有35TRACKS）。
6	IBSECT	被處理的磁區編號存放於此（0到15）。必須爲$00到$0F之間。
7 & 8	IBDCTP	這二個BYTE必須包含設備特性表的起始位址，BYTE 7包含低順序位址的BYTE，而BYTE 8包含高順序位址的BYTE。
9 & 10	IBBUFP	位元組9及10必須包含"資料緩衝區"的起始位址。這一資料緩衝區是RWTS副程式可處理的256 BYTE長度之記憶體部份。如果你要將資料寫到磁碟片去，那麼寫到磁碟片上去的，就是資料暫存區的資訊。如果你是從磁碟片上讀取資料，那麼從磁碟機取得的資訊將被存到記憶體資料緩衝區的位置去。
11 & 12	………………	沒有使用
13	IBCMD	在這個BYTE內存放著RWTS副程式到底在做什麼事情的命令代碼，存在BYTE 13的值有$00—無意義的命令，只做爲磁碟機儲存位置的開端而已。 $01—讀取存在磁碟片上指定的磁軌及磁區內整個256個BYTE的資料，且把這些資料存在記憶體的資料緩衝區位置上。 $02—將下一個儲存在記憶體內資料緩衝區位置上的256個BYTE的資料寫到磁碟片上指定的磁軌及磁區去。
14	IBSTAT	這個位置含有在整個執行情況下可能產生任何錯誤的代號編號，如果RWTS副程式產生進位旗標（CARRY FLAG）的清除則表示沒有發生任何錯誤，如果產生了進位旗標的設定，則這一個一BYTE將會指示出發生了何種錯誤。 $10—表示該磁碟片爲一保護住的唯讀磁片，無法將資料寫上去。 $20—磁册號碼無法匹配的錯誤。這個磁册號碼表示磁碟片被發現與BYTE 4所指定的磁册號碼不同。 $40—磁碟機發生錯誤，一些不正常的事情可能產生。 $80—讀取時發生錯誤。RWTS在經過48次重覆企圖讀取資料之後，仍然無法讀到位址資料欄（ADDRESS FIELD）或資料欄（DATA FIELD）。如果在指定的磁區資料欄內並沒有寫東西上去，這時就會產生一個讀取錯誤訊息，因爲這一位置上並無任何東西可讀取。
16	IOBPSN	這一位元組必須包含最近被處理到的插座編號乘上16後的值。例如：如果你剛處理到在插座6上的磁碟機，則存在此BYTE內的值卽爲$50。如果在指定的插座上沒有插置控制面板，則磁碟機將會停住。
17	IOBPDN	這個BYTE必須包含最近被處理到的磁碟機編號——爲$01或02。

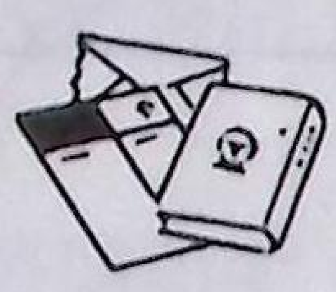
香港教育司署立案（1957年創校）

世界函授學校

規模大　優點多　科目好

講義
全部經教育司署審定，閱讀容易，印刷精美，免費供應。

師導
國內外著名大學學士，經香港教育司考核合格，教導認真，決不馬虎。

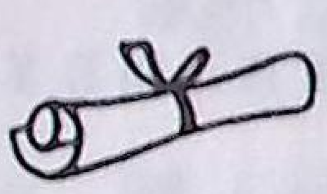




費用
除低廉之學費外，學生決無額外負擔。

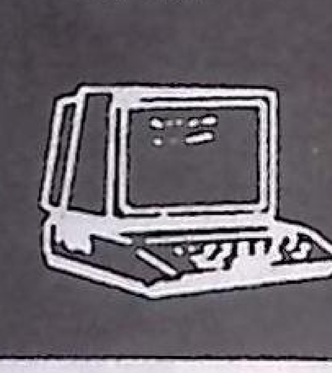
文憑
本校頒發文憑，均經政府簽証，實是最佳學歷証件。



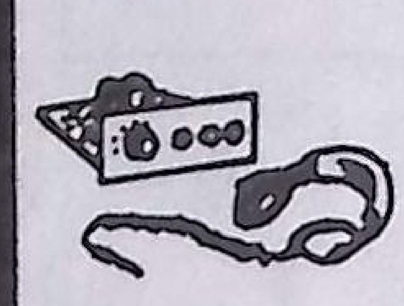


校址： 香港九龍彌敦道五六六號9樓A座
電話：③—803532

初級電腦科
電腦基本原理，電腦操作，邏輯電路，程式編寫八個月畢業。



初級無綫電科
全部圖解教導，原理裝修並重，極易學習，十個月畢業。



半導體裝修科
配合進步的技術，具初學或職業的裝修人員必修的學科十個月畢業。



實用數理
可以加强數理基礎費用少，收效大，6個月畢業。

x+y=z

實用商科
著重實際應用，加强商業學識，期間短，學習易六個月畢業。



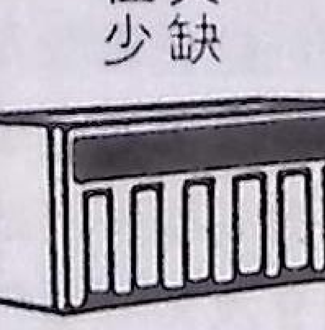
無綫電工程
設計，研究和深造的理想學科。一年半畢業。



汽車工程
適合職業及業餘人員加强對汽車方面之學識。



冷氣工程
冷氣，冷藏人員缺乏，專門學校極少一年半畢業。



商業美術科
內容新，範圍廣，包括美術和最新廣告設計。一年畢業。



應用漫畫
漫畫可訓練頭腦，增强思想，取材新，學習易，六個月畢業。



本港學生可利用電話詢問索取章程　電話③－803532

請依綫剪下，填好姓名，回郵地址，貼在信封上投寄，本校即免費寄上簡章

郵件請寄：P.O.BOX 70011, KOWLOON, CENTRAL POST OFFICE
九龍中央郵政局信箱70011號



技術就是財產

（寄自）
姓名：
地址：
（請註明要學的科目）
☐初級電腦科
☐無綫電工程科
☐初級無綫電科
☐半導體裝修科
☐實用數理科
☐汽車工程
☐冷氣工程
☐實用商科
☐商業美術
☐應用漫畫

請貼上郵票

香港教育司署立案

To：世界函授學校註册處

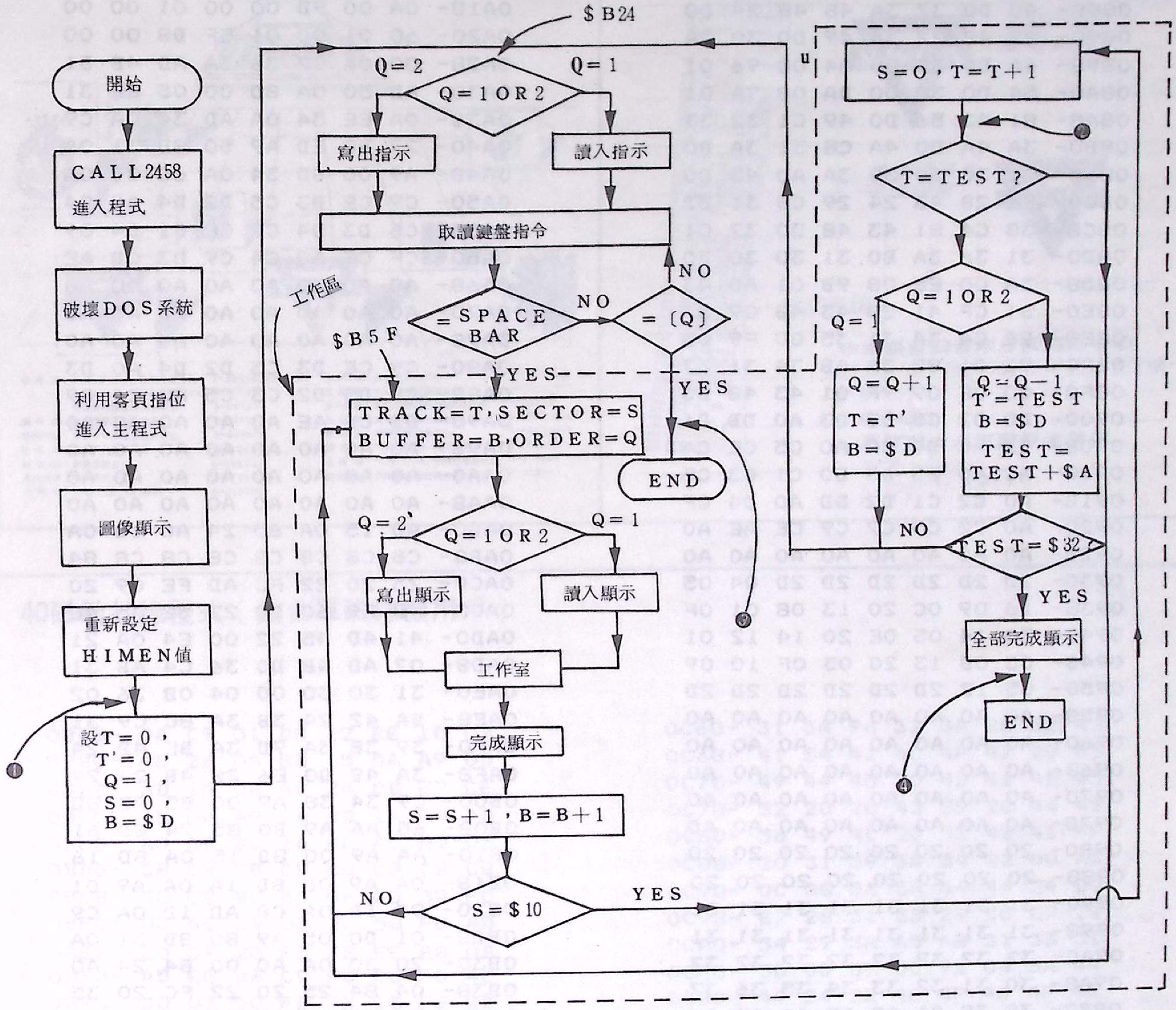
P. O. BOX No. 70011
KOWLOON CENTRAL POST OFFICE,
HONG KONG.

香港九龍中央郵局信箱70011號

電話：K-803532

學問就是力量

40磁軌抄錄程式(適合單磁碟機用戶)




```
REM ******************
REM ** EVERRY TEN **
REM **   COPIER   **
REM **     BY     **
REM ** SHAO YING  **
REM **    KIT     **
REM ******************
0800- 20 58 FC AD 00 09 8D 00
0808- 04 EE 04 08 EE 07 08 AD
0810- 07 08 C9 28 D0 ED A9 00
0818- 85 24 A9 03 85 25 20 22
0820- FC AD 30 09 20 FD FB EE
0828- 22 08 AD 22 08 C9 D6 D0
0830- F0 A9 20 8D D0 07 EE 34
0838- 08 AD 34 08 C9 FF D0 F1
0840- 4C 03 0B 00 2C 4A 29 29
0848- C8 31 32 38 00 53 08 CD
0850- 00 82 00 83 08 90 01 44
0858- 24 D0 E7 28 34 29 3A 97
0860- 3A BA 44 24 22 43 41 54
0868- 41 4C 4F 47 22 3A 42 D0
0870- E2 28 33 37 29 C9 32 3A
0878- AD 42 CF 32 32 C4 42 D0
0880- 20 00 0A 4C 00 08 01 43
```

40磁軌抄錄程式（適合單磁碟機用戶）

上文寫漏了一些東西，我說當破壞 DOS 系統的找尋 POINTER 的方法是說零頁的位址設爲零，而POINTE—R的指向通常是由位址 0 至 1F ，所以只要將位址0至1F設定爲零便可，若將全部零頁位址設定爲零，可能有意想不到的錯誤發生。

```
0888- 48 D0 37 3A 48 48 24 D0
0890- 22 48 22 3A 49 D0 30 3A
0898- 4A D0 30 00 A4 08 96 01
08A0- 54 D0 30 00 DA 08 9A 01
08A8- 81 43 56 D0 49 C1 32 33
08B0- 3A 4A D0 4A C8 31 3A B0
08B8- 31 30 30 30 3A AD 43 D0
08C0- E6 28 48 24 29 C8 31 32
08C8- 38 C4 81 43 48 D0 37 C1
08D0- 31 36 3A B0 31 30 30 30
08D8- 3A 00 EE 08 9B 01 AD 43
08E0- D1 CF 41 28 43 48 C9 37
08E8- 29 C4 34 31 35 00 F9 08
08F0- 9C 01 82 3A AB 34 31 37
08F8- 00 0F 09 9F 01 43 48 D0
0900- D0 D2 C5 D3 D3 A0 DB D1
0908- DD A0 D4 CF A0 C5 CE C4
0910- AC A0 DB D3 D0 C1 C3 C5
0918- A0 C2 C1 D2 DD A0 D4 CF
0920- A0 C2 C5 C7 C9 CE AE A0
0928- A0 A0 A0 A0 A0 A0 A0 A0
0930- 2D 2D 2D 2D 2D 2D 04 05
0938- 16 09 0C 20 13 08 01 0F
0940- 20 14 05 0E 20 14 12 01
0948- 03 0B 13 20 03 0F 10 09
0950- 05 12 2D 2D 2D 2D 2D 2D
0958- A0 A0 A0 A0 A0 A0 A0 A0
0960- A0 A0 A0 A0 A0 A0 A0 A0
0968- A0 A0 A0 A0 A0 A0 A0 A0
0970- A0 A0 A0 A0 A0 A0 A0 A0
0978- A0 A0 A0 A0 A0 A0 A0 A0
0980- 20 20 20 20 20 20 20 20
0988- 20 20 20 20 20 20 20 20
0990- 31 31 31 31 31 31 31 31
0998- 31 31 31 31 31 31 31 31
09A0- 32 32 32 32 32 32 32 32
09A8- 30 31 32 33 34 35 36 37
09B0- 38 39 01 02 03 04 05 06
09B8- 30 31 32 33 34 35 36 37
09C0- 38 39 01 02 03 04 05 06
09C8- 30 31 32 33 34 35 36 37
09D0- A0 A0 A0 A0 A0 A0 A0 A0
09D8- A0 A0 41 24 D0 EA 28 41
09E0- C9 D4 A7 D3 A0 C4 CF CE
09E8- C5 AE A0 A0 A0 A0 A0 A0
09F0- A0 A0 A0 A0 A9 4C 85 08
09F8- A9 80 85 09 A9 08 85 0A
0A00- A9 0A A0 11 20 B5 B7 60
0A08- 60 00 00 0A 60 60 60 00
0A10- 29 01 60 01 00 11 00 22
0A18- 0A 00 9D 00 00 01 00 00
0A20- 60 01 00 01 EF D8 00 00
0A28- 00 0A 37 36 3A AD 4B D1
0A30- AD 50 0A 8D 00 05 EE 31
0A38- 0A EE 34 0A AD 34 0A C9
0A40- 27 D0 ED A9 50 8D 31 0A
0A48- A9 00 8D 34 0A 60 81 0A
0A50- C9 CE D3 C5 D2 D4 A0 C4
0A58- C5 D3 D4 C9 CE C1 D4 C9
0A60- CF CE A0 C4 C9 D3 CB AE
0A68- A0 A0 A0 A0 A0 A0 A0 A0
0A70- A0 A0 A0 A0 A0 A0 A0 A0
0A78- A0 A0 A0 A0 A0 A0 A0 A0
0A80- C9 CE D3 C5 D2 D4 A0 D3
0A88- CF D5 D2 C3 C5 A0 C4 C9
0A90- D3 CB AE A0 A0 A0 A0 A0
0A98- A0 A0 A0 A0 A0 A0 A0 A0
0AA0- A0 A0 A0 A0 A0 A0 A0 A0
0AA8- A0 A0 A0 A0 A0 A0 A0 A0
0AB0- AD 15 0A 85 24 AC 16 0A
0AB8- C8 C8 C8 C8 C8 C8 C8 84
0AC0- 25 20 22 FC AD FE 09 20
0AC8- FD FB 60 D0 22 52 45 4E
0AD0- 41 4D 45 22 00 E4 0A 21
0AD8- 02 AD 4B D0 36 C4 AB 31
0AE0- 31 30 30 00 04 0B 26 02
0AE8- BA 42 24 3B 3A 8C C9 31
0AF0- 39 38 3A 9D 3A BE 4B 24
0AF8- 3A 4B D0 E6 28 4B 24 29
0B00- C9 34 38 A9 00 85 73 8D
0B08- 60 AA A9 B0 85 74 8D 61
0B10- AA A9 00 8D 15 0A 8D 16
0B18- 0A A9 0D 8D 1A 0A A9 01
0B20- 8D 1D 0A C8 AD 1D 0A C9
0B28- 01 D0 05 A9 80 8D 31 0A
0B30- 20 30 0A A0 00 84 24 A0
0B38- 04 84 25 20 22 FC 20 35
0B40- FD 8D FF 09 AD FF 09 C9
0B48- A0 F0 07 C9 D1 D0 E4 4C
0B50- D0 03 4C 5F 0B 56 D0 53
0B58- C9 54 C8 4B C9 31 00 AD
0B60- 1D 0A C9 01 F0 0B A9 D7
0B68- 8D FE 09 20 B0 0A 4C 79
0B70- 0B A9 D2 8D FE 09 20 B0
0B78- 0A 20 00 0A A9 AE 8D FE
0B80- 09 20 B0 0A EE 16 0A EE
0B88- 1A 0A AD 16 0A C9 10 F0
0B90- 03 4C 5F 0B A9 00 8D 16
0B98- 0A EE 15 0A AD 15 0A C9
0BA0- 0A F0 03 4C 5F 0B AD 1D
```

40磁軌抄錄程式（適合單磁碟機用戶）

```
0BA8- 0A C9 01 D0 12 EE 1D 0A
0BB0- AD 26 0A 8D 15 0A A9 0D
0BB8- 8D 1A 0A 4C 24 0B E7 CE
0BC0- 1D 0A A9 0D 8D 1A 0A AD
0BC8- A0 0B 8D 26 0A AC A0 0B
0BD0- C8 C8 C8 C8 C8 C8 C8 C8
0BD8- C8 C8 8C A0 0B C0 32 F0
0BE0- 03 4C 24 0B 20 58 FC AD
0BE8- E0 09 20 FD FB EE E8 0B
0BF0- AD E8 0B C9 F0 D0 F0 4C
0BF8- D0 03 00 FD FB EE F8 0B
0C00- AD F8 0B C9 F0 D0 F0 4C
0C08- D0 03 00 00 43 56 CB 38
0C10- 29 3A 43 32 D0 43 56 C9
0C18- 43 31 CA 38 3A 50 D0 31
0C20- 30 32 34 C8 31 32 38 CA
0C28- 43 32 C8 34 30 CA 43 31
0C30- C8 43 48 3A 43 D0 E2 28
0C38- 50 29 3A B1 00 57 0C 4C
0C40- 04 B2 20 43 41 54 41 4C
0C48- 4F 47 20 48 49 44 45 20
0C50- 4D 41 4B 45 52 22 00 8F
0C58- 0C 56 04 97 3A 9E 3A A2
0C60- 31 3A 96 31 3A BA 22 43
0C68- 41 54 41 4C 4F 47 20 48
0C70- 49 44 45 20 4D 41 4B 45
0C78- 52 20 28 43 29 20 44 45
0C80- 56 49 4C 20 53 48 41 4F
0C88- 20 31 39 38 34 22 00 AA
0C90- 0C 88 04 3A 44 44 24 D0
0C98- E7 28 31 33 29 C8 E7 28
0CA0- 34 29 3A A5 AB 31 33 31
0CA8- 30 00 D9 0C 92 04 BA 44
0CB0- 44 24 22 4E 4F 4D 4F 4E
0CB8- 43 49 4F 22 3A BA 22 28
0CC0- 2E 2E 2E 48 49 44 45 2E
0CC8- 2E 2E 29 22 3A 81 49 D0
0CD0- 31 C1 31 30 30 30 3A 82
0CD8- 00 EB 0C 9C 04 BA 44 44
0CE0- 24 22 43 41 54 41 4C 4F
0CE8- 47 22 00 2D 0D A6 04 BA
0CF0- 3A 9E 3A BA 46 49 24 3B
0CF8- 45 24 3A FB D9
*
```

確達64型電腦試機報告

雅式



機身內置64KＢ可編程序記憶、高解像度畫面配合16種不同顏色及活動圖案（SPRITES），三聲道音響合成器，6510先進中央處理，再加上一系列廉價的附件，相信足以吸引無數的電腦愛好者。確達64型（COMMODORE 64）電腦，一部早已瘋魔歐美的私人／家庭電腦，最近在香港推出。爲了解釋此機的性能，筆者特別粲集有關資料向各位讀者報道。

確達商業機器

確達商業機器是少數自行生產集成電路的微電腦製造商，在美國的知名度與另一家專門生產集成電路及電腦的公司——德州儀器有限公司並駕齊驅。確達的發達，可追溯到1956年紐約的一間打字機修理店的成立，經過多年的努力，至今乃已有近三十年的歷史。著名的中央處理器6502　，亦是由確達的附屬公司MOS　TECHNOLOG所發展。

確達電腦早期在港較爲人所熟知的型號有PET 8032及VIC—20 等。其中VIC—20是屬於初學者電腦，在歐美推出時亦曾引起搶購熱潮，惜在港推出時售價略高，加上AP Ⅱ出現，使其在港的銷量未能與歐美等地相提並論。

基本硬件

確達64型電腦的基本硬件體積只有一部手提打字機般大小，外型和昔日的VIC—20相似，但是機身的標貼和顏色有些不同，隨機有一個黑色的電源供應器，一條接電視接駁線及一本170 頁厚的用戶說明書。機面是一個相當濶大的鍵盤，電源顯示燈在機面的右上面；在機的背後，由左至右計，分別是已編程序盒帶揷口，電視訊號輸出口，外圍設備接口，錄音機接口及用戶埠等。在機的右側面分別是電源輸入，電源開關及控制桿１及２的揷口。全機的外型設計可以說是相當淸潔。

內部結構

一般人無論需要與否，都十分崇尙大記憶量的電腦，其實，以一般情形來說，有16KＢ記憶容量的電腦，已足夠一般初學者使用。確達64型電腦由於在設計時考慮到用戶將來的需要，所以標準的內存記憶容量由往日VIC—20的５Ｋ增加至64K，一般人或許以爲這與時下的一般64Ｋ電腦沒有甚麽分別，其實內裏結構有一個很顯著的分別，就是一般電腦當接上磁碟機後（包括APPLE Ⅱ）都會用十數ＫＢ的記憶，作爲儲存磁碟的操作系統程式。在確達64型電腦上面，由於磁碟操作系統是藏在ＲＯＭ裏面，所以其磁碟機之操作系統程式不會佔用電腦本體的ＲＡＭ。

８BIT的6510微處理器，和時下的6502 微處理使用同一系列的機械指令，但6510多了附加的８BIT輸入／輸出埠，使6510在同一時間內可選擇及處理超過84ＫＢ的記憶資料。同時，由於其機械指令和6502 一樣，所以大部份用6502 操作之程式，如APPLE Ⅱ，ATARI 等PROGRAM，均可經輕微改寫後，用於確達64。相反來說，由於6510的先進設計，確達64的程式却未能輕易改寫於其他機種裏應用。

圖像及音響

確達64型電腦選用比VIC—Ⅱ更強的6567作圖像介面集成電路，此枚6567除分担6510的圖像工作外，還提供一個320×300的高解像度畫面及活動圖案（SPRITE）。至於音响方面，它採用6581集成電路作音樂合成及聲响效果控製，使其可提供三聲音道，九組音階變化及ADSR（ATTACK，DECAY，SUSTAIN，RELEASE）控制以產生不同樂器之音色。

其他配件

確達64爲配合擴展用途，配件有控制桿，轉輪控製器，光筆，調制解調器，RS—232Ｃ介面咭，1EEE 488 介面咭，CENTRONICS并行介面咭等式式俱備；此外其更可透過介面咭接駁確達廠的商用外圍設備配件如硬磁碟機等。

總結

目前在港流行的家庭／私人電腦分別有假APPLE Ⅱ，ATARI及SPEC—TRAVIDEO ，而佔全港微電腦市塲最大的機種相信是有大量廉價配件及軟件之假APPLE Ⅱ居首。而ATARI 及SPECTRAVIDEO 則分別因價昂及軟件少成爲該等機種在港推銷的絆脚石。今天，COMMDORE 64 除有龐大而價廉的原廠或兼容外圍配件及軟件外，更具有先進的電子線路設計，使其在聲色上的表現均較同型機種爲優越，就單以解像度及音响而言，更能與IBM PC比美。相信此等優點亦是其奪得「全年最佳電腦獎」及瘋魔歐美各地之主要因素，而COMMODORE 64 亦可望在香港微電腦市塲中佔有一定地位。

改良WORDSTAR加强編印字體效果

蘋果長老

在衆多的文字處理器中，雖然 WORDSTAR 是功能最優異的一個，但要完全掌握它却需要花一些時間。不過它所編印的特殊效果比其它程式遠爲優勝，但需要更改印字機的用家位址內容。

筆者不喜歡它的 UNDERLING方法，因此希望改爲這種 UNDERLINE 形式，而且筆者的印字機沒有雙色功能，CONTROL PY這個換帶命令對筆者來說是沒有用的，相信大多數讀者都是如此。

爲了使字體更爲美觀出來，於是斜體字，放大及加寬編印模式便極爲需要，爲此要對WORDSTAR 加以修改，以下是其步驟：

一、將 CP／M MASTER 置放在A磁碟機上，然後將一張空磁片放在B機中。

二、用 FORMAT B：命令將空碟格式化，之後按

改良後的WORDSTAR所能編印出的字體模式

WORDSTAR

PRINT CONTROL COMMANDS

by Raymond

This document decribes the print control commands which are patched for Panasonic KX-P-1091E printer.

***** EXAMPLES OF PRINT CONTROL COMMANDS *****

PRINTOUT

ELITENORMAL
ELITERESET
NORMALRESET
CONDENSEDRESET
CON.+ENL.RESET
SUPERSCRIPTSRESET
SUPERSCRIPTSRESET
SUBSCRIPTSRESET
SUBSCRIPTSRESET
ENL.SUPERRESET
ENL.SUPERRESET
ENL.SUBRESET
ENL.SUBRESET
CON.SUPERRESET
CON.SUBRESET
C.+E.SUPERRESET

PRINTOUT

改良WORDSTAR加强編印字體效果

COPY B：＝A：／S將 CP／M 系統寫到新碟上。

三、移去 CP／M MASTER 並挿入 WORDSTAR 磁碟到A機上，將 WORDSTAR 複寫到B機上，按COPY B：＝A：

四、移去A，B機上磁片，B機上的磁片將被修改爲改良的 WORDSTAR 。

五、將 CP/M MASTER 放入A機上，再放一片空磁片到B機，利用 FORMAT B： 將它格式化。

六、按 PIP B:DDT.COM＝A:〔RET〕將DDT複寫到B機磁片上。

將先前所做的WORDSTAR 磁片放到A機，按入 PIP B:WS．COM＝A:〔RET〕把WORDSTAR 複寫到新碟上。

八、現在我們將要以 DDT 程式來修改 WORDSTAR 了，按B：〔RET〕再按 DDT B:WS.COM〔RET〕，這樣做會將DDT 及WS.COM 載入記憶體內。

九、當螢幕上出現 DDT 的符號“一”後，按入 S06C9〔RET〕，便會出現 00 緊隨在06C9之後。

十、06C9 這個位址是給予用家自設的五個位址中的第一個。

十一、當文件上按入 CTRL PQ時會檢查這個BYTE並告訴程式之後有那些位元要送到印字機上，這個數字原本是零。

十二、我們要利用這個位元來開啓放大編印模式，對EPSON相融的印字機來說需要一個 SHIFT OUT（SO）字符來開啓放大模式，因此 06C9這個位址要改爲 01，而下一個位址要改爲 0E（即 SO的十六進制代碼）。

這個我們先按入1〔RET〕，再按 0E〔RET〕

十三、按三次〔RET〕到 06CE位址上，這是USRZ的開始位址，按入1〔RET〕，再按 14〔RET〕。

十四、現在 CTRL PQ會開啓放大編印模式而CTRL PW 會關掉它。

十五、我們再按三次〔RET〕到 USR 3位址上（即06D3位址），這裏是擴張模式（EMPHASISED MO—DE）的開始位址，我們要兩個 ESCAPE 2來控制開啓。

十六、按2〔RET〕再按 45〔RET〕。

十七、再按〔RET〕兩次到 USR 4 位址上，位址是 06D8。

十八、我在USR 4 之上是存放 ESCAPE F ，這樣可以把 EMPHASISED MODE 關掉，因此按入 2〔RET〕。

十九、當下一個位址出現時按1B〔RET〕，下一個按46〔RET〕。

二十、以上所用的位址是WORDSTAR 提供予用家修改的，因此對 WORDSTAR 本身不會產生任何不良效果。

二十一、現在我們可以離開 DDT 並貯存起改良過的 WORDSTAR ，按入“·”在下一個位址並〔RET〕，這樣便會離開 DDT 的更改模式並給回“一”符號。

二十二、按入 CTRL C 離開 DDT ，這時會出現B＞的 CP／M符號。將你改良過的 WORDSTAR 貯存起來，以 NWS.COM命名之，以 SAVE 67B：NWS.COM 貯起。

二十三、按入A：回到A機上，將B機上的 WORDS—TAR 複寫到A機上，按入PIP A：WS.COM＝NWS COM〔RET〕。

二十四、這時的WORDSTAR 已包括了 ENLARGE．EMPHASISED 模式了，而且你可以用同樣方法來修改其他的功能，若果日後筆者有更好的改良時會再告訴大家，再見。■

捉蟲訓練——SINE WAVE

今期的捉虫問題與數學有關，是繪劃一條SINE曲綫，程式執行時沒有任何語法錯誤，可是出來的結果則有問題，問題發生在何處呢？

```
10  REM  THEW SINE WAVE
20  TEXT : HOME
30  PI = 3.14159
40  FOR X = 0 TO 10 STEP 2
50  Y =  SIN (PI * X / 20) * X
60  PRINT "!";: HTAB (10 - Y): PRINT
     "*";: HTAB (10 + Y): PRINT "
     *"
70  NEXT
80  PRINT "______________________
     __________________": REM  39
     DASHES
90  FOR X = 10 TO 1 STEP  - 2
100 Y =  SIN (PI * X / 20) * X
110  PRINT "!";: HTAB (30 - Y): PRINT
     "*";: HTAB (30 + Y): PRINT "
     *"
120  NEXT : END
```

我們鼓勵讀者自己設計一些「有蟲」的程式寄給我們，捉蟲訓練是公開給讀者的。程式長度沒有限制，當然越精簡越好，程式必需是用 APPLESOFT 語言，並需符合下列各項要求：

1、程式必須是 APPLESOFT ，任何 POKE 機械語言的子程式或 CALL 該子程式必須正確。

2、程式必須能出現有某些結果，而不會出岔子，換言之絕不能有 SYN—TAX ERROR出現。

3、程式必須附加簡短說明。

4、必須連同答案寄來。

假如閣下投寄來的捉蟲訓練程式是有水準，我們將贈閱十二期雜誌給你，或者再多一年的免費讀訂。

投稿請在信封面寫上「投稿捉蟲」。■

SINE曲綫問題答案：

問題出現於第40行，萬勿將HTAB設定爲0，大家可將第40行改爲：

40 FOR X＝1 TO 10 STEP 2

家庭開支及個人消費管理系統

無論私人，家庭及小型公司都適合採用的電子賬簿

本系統適合家庭、個人或小型公司使用，功能主要是記錄每項開支，系統雖然是以每週爲一單位，能在短時間內將過去一年或是財政年度的開支紀錄找出來及編印成一份報表。系統的優點在於用者可自行設定開支欄的名稱，一共有十六個。

無論是個人、家庭或私人公司都會發覺本系統的實用價值。

系統在使用形式設計上，非常接近VISICALC，此擧是方便大家日後開級使用VISICALC時有一預備。當然本系統的速度跟VISICALC可以相提並論，而且在使用上十分簡單，只要十分鐘卽學會。此亦是本系統的一大特點。

程式能將一週內的開支作詳細記錄，並能卽時編印一份硬本以便查核存案。同時，用者可將每週的開支紀錄收錄入一張資料碟，以便能有一延續的開支消費報告，編印一份的YEAR TO DATE 開支報表。由於螢幕顯示的限制，每項開支最大的數值被限定在999.99元以下。而YEAR TO DATE單項開支最大數額亦不能超過99,999.99元。用家必須留意。筆者相信，這不會造成不便。

當執行本系統後，螢幕會出現如圖(一)的畫面。用家可選擇輸入資料或返回主索引。

主索引共有七項選擇（見圖2）。至於十六項開支欄名，用家可視乎自己要求而定。這裏，我列出如下：

(1)	BOOKS／MAGAZINE	雜誌報紙
(2)	BREAKFAST	早餐
(3)	LUNCH	午餐
(4)	DINNER	晚膳
(5)	CINEMA	電影娛樂
(6)	CLOTHES	服裝
(7)	CIGARETTE	香煙
(8)	FOOD	食物
(9)	GAS	煤氣
(10)	WATER／ELECTRICITY	水電費
(11)	MORTAGE	供屋
(12)	PHONE	電話
(13)	SCHOOL FEE	學費
(14)	TRANSPORT	交通
(15)	HOBBY	娛樂
(16)	OTHER	其他

當然，要更改這些欄名字非常容易，大家只要改動程式中行句11440至11740便可。

由於程式在設計時要遷就小型公司的應用，故在欄名之下有兩項是專爲此而設：

其一是COMPANY CHARGE （稱寫CO CHGES）。有些花費開支是爲公司賬，故可從SUBTOTAL中扣除，而得出一個NET TOTAL。

另一個是ADVANCE，用途是公司可能事先預付一筆現金給你，換句話說，你便有一筆可調動的資金，在結算時應自動減除。

ADVANCE 的數值是在行句1990及11240中設定，這裏設定爲0，卽11240 AD＝0

如果你預支付有500元，可將此值改爲11240 AD＝500

```
  ITEM         SUN      MON      TUE      WED
  ----         ---      ---      ---      ---
 1 BOOK/MAG
 2 BREAKFST
 3 LUNCH
 4 DINNER
 5 CINEMA
 6 CLOTHES
 7 CIGRETTE            HOME EXPENSE-CALC
 8 FOOD                BY
 9 GAS                 ALBERT NG
10 WAT/ELEC
11 MORTAGE
12 PHONE               PLEASE CHOSE ONE:
13 SCL FREE            <1> INPUT DATA
14 TRANSPRT            <2> MAIN MENU
15 HOBBY               ->
16 OTHER
SUBTOTAL ->
- CO CHGES.
NET TOTAL->
ADVANCE
DIFFERENCE
INITIAL SET UP
```

圖(1)

本系統的操作非常簡易。螢幕下部會有操作指令簡介。由於系統是使用40字行顯示，故在一螢幕上只能分爲兩部顯示。用家利用L及R鍵來觀看這兩部份。左右箭咀是用來移動星期欄。輸入數額可用RETURN，要改錯重新輸入資料或返回上一個開支消費欄可用ESC．Q鍵是用來離開系統。

由於本系統是以一週爲單位。故在第七日卽週末輸入完所有資料後，應當編造一份完整的報表。

當你要將一週的財政報告SAVE時，電腦會自動將有關數值輸入YEAR TO DATE （YTD） 檔案，方便日後做年結統計用。在SAVE檔案時留意不要輸入相同日期的檔案名字，否則會將以前的資料覆蓋。

用家可利用PRINTER將紀錄編印成一份硬本供備案。

本系統程式同分爲兩部：HOME—EXPENSECALC及AMPER—INTEPRETER。後者程式是爲了要讓系統工作速度加快。

由於本系統所有的使用指示均在螢幕上有清楚的指出，故不擬在此多說。■

家庭開支及個人消費管理系統

```
10   REM **********************
11   REM *  HOME EXPENSE-CALC *
12   REM *  BY ALBERT  NG     *
13   REM * COPYRIGHT (C) 1984 *
14   REM * BY COMPUTING AGE   *
15   REM *                    *
16   REM **********************
1190  POKE 768,6: POKE 769,1
1200  HOME : PRINT "EC": HTAB 10:
   VTAB 12: INVERSE : PRINT "O
  NE MOMENT PLEASE": NORMAL : VTAB
   20: PRINT "** COPYRIGHT 1984
   BY COMPUTING AGE LTD**"
1220 D$ =  CHR$ (4)
1240  PRINT D$"BRUN AMPER-INTERPR
  ETER"
1260  GOTO 11000
1280  REM ----------------------
1300  REM SUB TO SET CURSOR POS
1320  REM ----------------------
1340  VTAB Y: HTAB X: RETURN
1360  REM ----------------------
1380  REM SUB TO PRINT DATA COLM
1400  REM ----------------------
1420 EX = 1:Y = 1:X = 16 + P * 7:
   GOSUB 1340: PRINT DA$(DA)
1440 Y = 2:X = X - 2: GOSUB 1340
1460  INVERSE : PRINT "------": NORMAL

1480  REM     PRINT DATA
1500  FOR Y = 3 TO 18
1520  GOSUB 1340
1540 OP = IT(DA,EX):
1560  & PRNT,OP,F$
1580 EX = EX + 1
1600  NEXT Y
1620  REM   PRINT SUBTOTAL
1640 X = X - 1:Y = 19: GOSUB 1340
   :OP = 0
1660  & PRNT,IT(DA,17),F2$
1680 X = X + 1
1700  REM PRINT PERSONAL MILEAGE
1720 Y = 20: GOSUB 1340
1740  & PRNT,IT(DA,18),F$
1760  REM    PRINT NET TOTAL
1780 Y = 21:X = X - 1: GOSUB 1340

1800  & PRNT,IT(DA,19),F2$
1820 X = X + 1: IF DA <  > 8 THEN
   RETURN
1840  REM IF TOTALS PRNT ADV/DIF
1860  IF IT(8,19) < 10000 GOTO 19
   40
1880 Y = 21:X = X - 3: GOSUB 1340

1900  & PRNT,IT(DA,19),F3$
1920 X = X + 3
1940 Y = 22: GOSUB 1340
1950  IF  LEFT$ (DA$,4) = "YEAR" THEN
   AD = HOLD(8,20)
1960  & PRNT,AD,F$
1970 DI = IT(8,19) - AD
1980 X = X - 3:Y = 23: GOSUB 1340
```

```
                    HOME EXPENSE-CALC

  OPTIONS:

1) INPUT A NEW REPORT

2) REVIEW/CHANGE/DELETE/UPDATE

3) SEE CATALOG OF THIS DISK'S FILES

4) PRINT HARD COPY OF DATA

5) CHANGE DATE/NAME OF A FILE

6) CHANGE DRIVE # : DRIVE NOW = 1

7) CHANGE SLOT # :  SLOT NOW = 6

8) END



PLEASE TYPE THE NUMBER OF YOUR CHOICE
```

圖(2)

```
1990 AD =  0 : REM  CUSTOMIZE TO
   YOUR OWN TASTE
2000  & PRNT,DI,F3$
2020  RETURN
2040  REM  **********************

2060  REM
2080  REM  SPACE RESERVED FOR A
2100  REM  BASIC SCREEN FORMAT-
2120  REM  ING ROUTINE IN CASE A
2140  REM  "PRINT USING" UTILITY
2160  REM  IS NOT AVAILABLE.  SEE

2180  REM  ARTICLE FOR OTHER RE-
2200  REM  QUIRED CHANGES.
2220  REM
2240  REM  **********************

2260  REM ----------------------
2280  REM ROUTINE TO INPUT DATA
2300  REM ----------------------
2320 A$ = "":E$ = "":X = 14 + P *
   7
2340  FOR Y = 3 TO 18: GOSUB 2360
   : NEXT : GOTO 2900
2360 EX = Y - 2:I$ = EX$(EX): GOSUB
   3280: GOSUB 3180:CT = 0: IF
   Y = 19 THEN  POP : GOSUB 330
   0: GOTO 2320
2380  IF EX = 1 THEN I$ = INST$: GOSUB
   3180
2400  GOSUB 1340
2420  IF IT(DA,EX) <  > 0 THEN  GOSUB
```

```
    1340: PRINT  CHR$ (8)">";: &
    PRNT,IT(DA,EX),F$: GOTO 2460
2440  PRINT  CHR$ (8)">"UNDERLINE
    $
2460  GOSUB 1340
2480 A$ = "":E$ = "":CT = 0: GOSUB
    1340
2500  POKE 49168,0: GET A$
2520  IF A$ = "L" OR A$ = "R" OR
    A$ =  CHR$ (8) OR A$ =  CHR$
    (21) OR A$ = "Q" THEN  POP :
     GOSUB 3300:IT(DA,17) = 0: FOR
     I = 1 TO 16:IT(DA,17) = IT(D
    A,17) + IT(DA,I): NEXT :IT(D
    A,19) = IT(DA,17) - IT(DA,18
    ):OP = IT(DA,19): GOSUB 1340
    : PRINT  CHR$ (8)" ";: RETURN
2540  IF A$ =  CHR$ (13) AND IT(D
    A,EX) < > 0 GOTO 2820
2560  IF A$ =  CHR$ (13) THEN E$ =
    "0": GOTO 2780
2580  IF A$ <  > CHR$ (8) AND A$
     <  >  CHR$ (21) AND A$ <  >
     CHR$ (27) AND A$ <  > "L" AND
    A$ <  > "R" AND A$ <  > "Q" AND
    A$ <  > "." AND (A$ < "0" OR
    A$ > "9") GOTO 2500
2600  GOTO 2640
2620  POKE 49168,0: GET A$
2640  IF A$ =  CHR$ (27) THEN  GOSUB
    3300: GOSUB 1340: PRINT  CHR$
    (8)" ";: & PRNT,IT(DA,EX),F$
    :Y = Y - 1: IF Y< 3 THEN Y =
    3
2660  IF A$ =  CHR$ (27) THEN  GOTO
    2360
2680  IF A$ =  CHR$ (13) GOTO 278
    0
2700  IF (A$ < "0" OR A$ > "9") AND
    A$ <  > "." THEN  GOTO 2360
2720  IF CT = 0 AND IT(DA,EX) <  >
    0 THEN  PRINT "       ";: GOSUB
    1340
2740  PRINT A$;:E$ = E$ + A$:CT =
    CT + 1: IF CT = 6 GOTO 2780
2760  GOTO 2620
2780 IT(DA,EX) = ( INT ( VAL (E$)
    * 100 + 0.001)) / 100
2800  IF IT(DA,EX) > 999.99 THEN
    I$ = "ITEM > $999.99": FLASH
    : GOSUB 3200: FOR J = 1 TO 2
    000: NEXT J:IT(DA,EX) = 0: GOTO
    2360
2820 X = X - 1: GOSUB 1340: PRINT
    " ";: & PRNT,IT(DA,EX),F$:E$
    = ""
2840  GOSUB 3300:X = X + 1
2860  RETURN
2880  REM  DO SUBTOTAL
2900 Y = 19: GOSUB 1340:OP = 0:IT
    (DA,17) = 0
2920  FOR I = 1 TO 16:IT(DA,17) =
    IT(DA,17) + IT(DA,I): NEXT :
    IT(DA,17) = ( INT (IT(DA,17)
     * 100 + 0.001)) / 100
2940  & PRNT,IT(DA,17),F$
2960  REM  GET PERSONAL MILEAGE
2980 Y = 20: GOSUB 2360: GOSUB 13
    40
3000  & PRNT,IT(DA,18),F$
3020  REM  PRINT NET TOTAL
3040 I$ = INST$: GOSUB 3180
3060 Y = 21: GOSUB 1340:OP = IT(D
    A,17) - IT(DA,18):IT(DA,19) =
    ( INT (OP * 100 + 0.001)) /
    100
3080  & PRNT,OP,F$
3100  RETURN
3120  REM -------------------
3140  REM PRINT INSTRUCTION LINE
3160  REM -------------------
3180  INVERSE
3200  VTAB 24: HTAB 1: CALL  - 86
    8: PRINT I$;: NORMAL : RETURN
3220  REM -------------------
3240  REM  PRINT ITEM NAME
3260  REM -------------------
3280  INVERSE
3300  XS = X:X = 1: GOSUB 1340
3320  PRINT EX$(EX): NORMAL :X =
    XS: RETURN
3340  GOSUB 1340
3360  PRINT EX$(EX)
3380  NORMAL :X = XS
3400  RETURN
3420  REM ---------------------
3440  REM SCREEN SET UP HERE
3460  REM ---------------------
3480  REM  SET UP SCREEN
3500  HOME
3520  PRINT "   ITEM        SUN
     MON    TUE   WED ";
3540  INVERSE
3560  PRINT "----            ----
    ---   ---    ";
3580  NORMAL
3600  FOR I = 1 TO 21
3620  PRINT EX$(I)
3640  NEXT
3660 I$ = "INITIAL SET UP"
3680  GOSUB 3180
3700  IF S% = 1 THEN  RETURN
3720 X = 20:Y = 9: GOSUB 1340
3740  PRINT "HOME EXPENSE-CALC"
3760 Y = 10: GOSUB 1340: PRINT "B
    Y"
3780 Y = 11: GOSUB 1340: PRINT "A
    LBERT NG"
3800 Y = 14: GOSUB 1340: PRINT "P
    LEASE CHOSE ONE: ";
3820 Y = 15: GOSUB 1340: PRINT "<
    1> INPUT DATA";
3840 Y = 16: GOSUB 1340: PRINT "<
    2> MAIN MENU ";
3860 Y = 17: GOSUB 1340: FLASH : PRINT
```

```
     "->";: NORMAL : PRINT " ";:
     GET A$
3880 S% = 1
3900  IF A$ = "1" GOTO 8940
3920  IF A$ = "2" THEN  PRINT A$:
     GOTO 8660
3940  GOTO 3860
3960  REM --------------------
3980  REM SUB TO REDUCE WINDOW
4000  REM --------------------
4020  POKE 33,29: POKE 32,11: POKE
     34,2: RETURN
4040  REM --------------------
4060  REM  REVIEW/CHG/DEL/UPDATE
4080  REM --------------------
4100  HOME :S2% = 0
4120  INVERSE : PRINT  SPC( 9)"RE
    VIEW/CHANGE/UPDATE" SPC( 10)
    : NORMAL : PRINT
4140  VTAB 8: PRINT "DO YOU WANT:
    ": PRINT : PRINT : PRINT "(1
    ) YEAR-TO-DATE INFORMATION":
     PRINT : PRINT "(2) A PAST W
    EEK'S EXPENSES": IF S4% = 1 THEN
     PRINT : PRINT "(3) THE DATA
     JUST INPUT"
4160  PRINT : PRINT : PRINT "PRES
    S ESC TO RETURN TO THE MAIN
    MENU OR INDICATE YOUR CHOICE
     BY NUMBER PLEASE: ";: GET A
    NS$: PRINT ANS$: IF ANS$ =  CHR$
    (27) GOTO 8660
4180  IF ANS$ < "1" OR ANS$ > "3"
     GOTO 4100
4200 A =  VAL (ANS$): PRINT : PRINT
    : PRINT OMP$: IF A = 3 AND S
    4% <  > 1 GOTO 4100
4220  ON A GOTO 4260,4480,4860
4240  REM YEAR-TO-DATE
4260  HOME
4280  IF S4% = 1 AND S1% = 0 THEN
     PRINT : PRINT "NOTE: THIS W
    ILL DESTROY THE DATA WHICH
    WAS JUST INPUT BUT NOT SAVED
    . PRESS ESC TO CONTINUE--ANY
    THING ELSE TO RETURN: ";: GET
    A$: PRINT A$: IF A$ <  >  CHR$
    (27) GOTO 4100
4300 DA$ = "YEAR-TO-DATE"
4320 S1% = 0:S2% = 1:S3% = 1:S4% =
    0:S5% = 1
4340  INVERSE : PRINT  SPC( 10)DA
    $" TOTALS" SPC( 11): NORMAL

4360  POKE 34,1
4380  GOSUB 6220
4400  TEXT
4420  FOR DA = 1 TO 8: FOR EX = 1
     TO 20:IT(DA,EX) = HOLD(DA,E
    X): NEXT EX,DA
4440  GOTO 4860
4460  REM  PAST WEEK'S EXPENSES
4480  HOME
4500  IF S4% = 1 AND S1% = 0 THEN
     PRINT "NOTE: THIS WILL DEST
    ROY THE DATA WHICH  WAS JUST
     INPUT AND NOT YET SAVED.  P
    RESSESC TO CONTINUE--ANYTHIN
    G ELSE TO START OVER. ->";: GET
    A$: PRINT A$: IF A$ <  >  CHR$
    (27) GOTO 4100
4520 S4% = 0:S2% = 1:S3% = 0: HOME

4540  INVERSE :TI$ = "      REVIEW
    A PAST WEEK'S EXPENSES
    ": PRINT TI$: NORMAL : POKE
    34,1
4560  VTAB 8: PRINT "DO YOU WANT
    A CATALOG (Y/N)?";: GET A$: PRINT
    A$: IF A$ = "Y" THEN  PRINT
    D$"CATALOG,S"SL",D"DR: PRINT
    : PRINT : PRINT "PRESS ANY K
    EY TO CONTINUE->";: GET B$: PRINT
    B$: GOTO 4600
4580  IF A$ <  > "N" GOTO 4560
4600  HOME : VTAB 12: PRINT "PLEA
    SE INPUT THE DATE/NAME OF TH
    E WEEK  WHICH YOU WISH TO RE
    VIEW.": GOSUB 7060
4620  HOME : PRINT OMP$
4640  PRINT D$"MON C,I,O"
4660  PRINT D$"OPEN EXPENSES FOR
    "DA$",L9,S"SL",D"DR
4680  FOR DA = 1 TO 8
4700  FOR EX = 1 TO 19
4720  PRINT D$"READ EXPENSES FOR
    "DA$",R";(DA - 1) * 19 + EX
4740  INPUT IT(DA,EX)
4760  NEXT EX,DA
4780  PRINT D$"CLOSE"
4800  PRINT D$"NOMON C,I,O"
4820  TEXT
4840  REM  REVIEW DATA ROUTINE
4860  GOSUB 3500
4880 P = 0:S = 1
4900  GOSUB 4020: HOME : TEXT :I$
     = "REVIEW: <-/->/L/R/C(HANG
    E/P(RINT/ESC": GOSUB 3180
4920  FOR DA = S TO S + 3: GOSUB
    1420:P = P + 1: NEXT :P = 0
4940 I$ = "REVIEW: <-/->/L/R/C(HA
    NGE/P(RINT/ESC": GOSUB 3180
4960  GET A$
4980  IF A$ =  CHR$ (8) THEN S =
    S - 1: IF S < 1 THEN S = 1: GOTO
    4960
5000  IF A$ =  CHR$ (8) GOTO 4900
5020  IF A$ =  CHR$ (21) THEN S =
    S + 1: IF S > 5 THEN S = 5: GOTO
    4960
5040  IF A$ =  CHR$ (21) GOTO 490
    0
5060  IF A$ = "L" THEN S = 1: GOTO
    4900
5080  IF A$ = "R" THEN S = 5: GOTO
    4900
5100  IF A$ = "C" AND ANS$ = "1" THEN
```

```
     I$ = "Y-T-D CHANGES NOT ALLO
     WED": FLASH : GOSUB 3200: FOR
     I = 1 TO 1200: NEXT : GOTO 4
     920
5120 IF A$ = "C" GOTO 5240
5140 IF A$ = CHR$ (27) GOTO 866
     0
5160 IF A$ = "P" AND S3% = 0 THEN
     GOSUB 7060
5180 IF A$ = "P" THEN GOSUB 922
     0: GOTO 4920
5200 GOTO 4960
5220 REM CHANGE ROUTINE
5240 HOME :S5% = 1
5260 IF S4% = 1 AND S1% = 0 GOTO
     5380
5280 GOSUB 6200: PRINT OMP$
5300 FOR DA = 1 TO 8
5320 FOR EX = 1 TO 19
5340 HOLD(DA,EX) = ( INT ((HOLD(D
     A,EX) - IT(DA,EX)) * 100 + .
     001)) / 100
5360 NEXT EX,DA
5370 HOLD(8,20) = HOLD(8,20) - AD
5380 GOSUB 3500: REM MAIN INPUT
          ENTERS HERE
5400 GOSUB 4020: HOME : TEXT :I$
     = INST$: GOSUB 3180
5420 DA = 1: FOR P = 0 TO 3: GOSUB
     1420:DA = DA + 1: NEXT :P =
     0:DA = 1
5440 GOSUB 2320
5460 I$ = INST$: GOSUB 3180
5480 FOR PAUSE = 1 TO 1000: NEXT
5500 GOSUB 1420
5520 IF A$ = "L" GOTO 5400
5540 IF A$ = "R" THEN DA = 5: GOSUB
     4020: HOME : TEXT :I$ = INST
     $: GOSUB 3180: FOR P = 0 TO
     3: GOSUB 1420:DA = DA + 1: NEXT
     :DA = 5:P = 0: GOTO 5440
5560 IF A$ = "Q" GOTO 5720
5580 IF A$ = CHR$ (8) THEN DA =
     DA - 1:P = P - 1: IF DA < 1 THEN
     DA = 1:P = 0
5600 IF P < 0 THEN GOSUB 4020: HOME
     : TEXT : FOR P = 0 TO 3: GOSUB
     1420:DA = DA + 1: NEXT :P =
     0:DA = DA - 4
5620 IF A$ = CHR$ (8) GOTO 5440
5640 DA = DA + 1:P = P + 1
5660 IF DA = 8 GOTO 5720
5680 IF P > 3 THEN GOSUB 4020: HOME
     : TEXT :DA = DA - 3: FOR P =
     0 TO 3: GOSUB 1420:DA = DA +
     1: NEXT :DA = DA - 1:P = 3: GOTO
     5440
5700 GOTO 5440
5720 I$ = OMP$: FLASH : GOSUB 320
     0: NORMAL : FOR EX = 1 TO 19
     :IT(8,EX) = 0: FOR DA = 1 TO
     7:IT(8,EX) = IT(8,EX) + IT(D
     A,EX): NEXT DA,EX
5740 DA = 5: FOR P = 0 TO 3: GOSUB
     1420:DA = DA + 1: NEXT
5760 POKE - 16368,0:I$ = "SELEC
     T: 1-DEL RPT 2-SAVE 3-CONT 4
     -MENU:": GOSUB 3180: GET A$:
     IF A$ < "1" OR A$ > "4" GOTO
     5760
5780 POKE 2039, ASC (A$) + 128
5800 X = VAL (A$)
5820 ON X GOTO 5920,5860,5400,86
     60
5840 REM SAVE RPT/UPDATE Y-T-D
5860 IF S4% = 1 AND S1% = 0 THEN
     S2% = 0:S5% = 0: POKE 34,23:
     HOME : PRINT : TEXT : GOTO
     6160
5880 HOME :S5% = 0: GOTO 6460
5900 REM   CLEAR & ERASE
5920 HOME
5940 IF S4% = 1 THEN PRINT "DAT
     A JUST INPUT HAS BEEN CLEARE
     D.": CLEAR :S% = 1: VTAB 10:
     PRINT "PRESS ANY KEY TO CON
     TINUE: ";: GET A$: PRINT A$:
     GOTO 11020
5960 PRINT "THIS FUNCTION WILL C
     OMPLETELY DELETE   THE REPO
     RT FILE TITLED:": PRINT : PRINT
     "EXPENSES FOR "DA$
5980 PRINT : PRINT "AND UPDATE T
     HE YEAR TO DATE TOTALS TO
     REFLECT IT'S REMOVAL.  IF YO
     U ARE SURE  YOU WISH TO CONT
     INUE PRESS ESC.  ANY   OTHE
     R KEY WILL RETURN YOU TO THE
      CHANGE MODE.  ->";: GET A$: PRINT
     A$
6000 IF A$ < > CHR$ (27) GOTO
     5380
6020 PRINT D$"MON C,I,O"
6040 PRINT D$"DELETE EXPENSES FO
     R "DA$",S"SL",D"DR
6060 S5% = 1: GOSUB 6620
6080 GOTO 8660
6100 REM -------------------
6120 REM UPDATE TOTALS FILE
6140 REM -------------------
6160 HOME : INVERSE : PRINT "GET
     TING YEAR-TO-DATE TOTALS FIL
     E": NORMAL
6180 PRINT : GOTO 6220
6200 HOME : INVERSE : PRINT "ONE
     MOMENT PLEASE": NORMAL
6220 PRINT D$"VERIFY EXPCALC-TOT
     ALS"
6240 REM IF NO ERR SAVE FILES
6260 HOME : PRINT : PRINT D$"MON
     C,I,O"
6280 PRINT D$"OPEN EXPCALC-TOTAL
     S,L9,S"SL",D"DR
6300 FOR DA = 1 TO 8
6320 FOR EX = 1 TO 19
```

```
6340  PRINT D$"READ EXPCALC-TOTAL
     S,R"(DA - 1) * 19 + EX
6360  INPUT HOLD(DA,EX)
6380  NEXT EX,DA
6390  INPUT HOLD(8,20)
6400  PRINT D$;"CLOSE EXPCALC-TOT
     ALS"
6420  PRINT D$"NOMON C,I,O"
6440  IF S5% = 1 OR S2% = 1 THEN
      RETURN
6460  HOME : INVERSE : PRINT " UP
     DATING YEAR-TO-DATE TOTALS F
     ILE DATA": NORMAL : PRINT : PRINT
     : PRINT OMP$
6480  PRINT D$"MON C,I,O"
6500  FOR DA = 1 TO 8
6520  PRINT DA$(DA)
6540  FOR EX = 1 TO 19
6560 HOLD(DA,EX) = ( INT ((HOLD(D
     A,EX) + IT(DA,EX)) * 100 + .
     001)) / 100
6580  NEXT EX,DA
6590 HOLD(8,20) = HOLD(8,20) + AD
6600  HOME : INVERSE : PRINT "WRI
     TING YEAR-TO-DATE TOTALS FIL
     E": NORMAL : PRINT : PRINT
6620  PRINT D$"OPEN EXPCALC-TOTAL
     S,L9,S"SL",D"DR
6640  PRINT D$"DELETE EXPCALC-TOT
     ALS,S"SL",D"DR
6660  PRINT D$"OPEN EXPCALC-TOTAL
     S,L9,S"SL",D"DR
6680  FOR DA = 1 TO 8
6700  FOR EX = 1 TO 19
6720  PRINT D$;"WRITE EXPCALC-TOT
     ALS,R";(DA - 1) * 19 + EX
6740  PRINT ( INT (HOLD(DA,EX) *
     100 + 0.001)) / 100
6760  NEXT EX,DA
6770  PRINT HOLD(8,20)
6780  PRINT D$"CLOSE"
6800  PRINT D$"NOMON C,I,O"
6820  IF S5% = 1 THEN  RETURN
6840  HOME
6860  INVERSE : PRINT "UPDATED TO
     TALS FOR YEAR-TO-DATE": NORMAL
     : PRINT
6880  FOR I = 1 TO 20
6900  VTAB I + 2: HTAB 1
6920  PRINT EX$(I); SPC( 10)
6940  & PRNT,HOLD(8,I),F3$
6960  NEXT : VTAB 23: HTAB 39: PRINT

6980  IF S3% = 1 THEN  VTAB 24: PRINT
     "PRESS ANY KEY TO CONTINUE:
     ";: GET A$: PRINT A$: HOME
7000  REM --------------------
7020  REM GET AND VERIFY DATE
7040  REM --------------------
7060  IF S3% = 1 GOTO 7700
7080 I$ = "ASSIGN DATE/NAME TO TH
     E EXPENSE REPORT": GOSUB 318
     0:DA$ = "":A$ = "":MO$ = "":
     YR$ = "":DY$ = "": GOSUB 714
     0
7100  IF S2% = 1 THEN  RETURN
7120  GOTO 7700
7140 X = 1:Y = 23: GOSUB 1340: CALL
      - 868: PRINT "INPUT # OF MO
     NTH (1-12) -> ";:A$ = "":MO$
     = ""
7160  GET A$: IF A$ >  CHR$ (47) AND
     A$ <  CHR$ (58) THEN MO$ = M
     O$ + A$: PRINT A$;: GOTO 716
     0
7180  IF A$ =  CHR$ (13) OR A$ =
     "/" THEN A$ = "/": PRINT A$;
     : GOTO 7220
7200  GOTO 7140
7220 DT% =  VAL (MO$): IF DT% < 1
      OR DT% > 12 GOTO 7140
7240 MO$ = MO$ + A$
7260  GOSUB 1340: CALL  - 868: PRINT
     "INPUT STARTING DAY DATE (1-
     31) -> "MO$;:A$ = "":DY$ = "
     "
7280  GET A$: IF A$ >  CHR$ (47) AND
     A$ <  CHR$ (58) THEN DY$ = D
     Y$ + A$: PRINT A$;: GOTO 728
     0
7300  IF A$ =  CHR$ (13) OR A$ =
     "/" THEN A$ = "/": PRINT A$;
     : GOTO 7340
7320  GOTO 7260
7340 DT% =  VAL (DY$): IF DT% < 1
      OR DT% > 31 GOTO 7260
7360 DY$ = DY$ + A$
7380  GOSUB 1340: CALL  - 868: PRINT
     "INPUT YEAR ->  "MO$DY$;:A$ =
     "":YR$ = ""
7400  GET A$: IF A$ >  CHR$ (47) AND
     A$ <  CHR$ (58) THEN YR$ = Y
     R$ + A$: PRINT A$;: GOTO 740
     0
7420  IF A$ =  CHR$ (13) GOTO 746
     0
7440  GOTO 7380
7460  IF  LEN (YR$) = 2 OR  LEN (
     YR$) = 4 GOTO 7500
7480  GOTO 7380
7500  IF  LEN (YR$) = 2 THEN YR$ =
     "19" + YR$
7520 A$ = " ": PRINT A$;:DA$ = MO
     $ + DY$ + YR$
7540  GOSUB 1340: CALL  - 868: PRINT
     DA$" <- IS THIS DATE OK (Y/N
     )?";: GET A$: IF A$ =  CHR$
     (13) THEN A$ = "Y"
7560  PRINT A$: IF A$ = "N" GOTO
     7140
7580  IF A$ <  > "Y" GOTO 7540
7600 S3% = 1
7620  RETURN
7640  REM --------------------
7660  REM  SEND TO DISK
7680  REM --------------------
7700  HOME
```

```
7720  PRINT D$"MON C,I,O"
7740 D$ =  CHR$ (4): PRINT D$;"OP
   EN EXPENSES FOR ";DA$;",L9,S
   "SL",D"DR
7760  PRINT D$;"DELETE EXPENSES F
   OR "DA$",S"SL",D"DR
7780  PRINT D$;"OPEN EXPENSES FOR
   ";DA$;",L9,S"SL",D"DR
7800  FOR H = 1 TO 8
7820  FOR I = 1 TO 19
7840  PRINT D$;"WRITE EXPENSES FO
   R ";DA$;",R";(H - 1) * 19 +
   I
7860  PRINT ( INT (IT(H,I) * 100 +
   0.001)) / 100
7880  NEXT I,H
7900  PRINT D$;"CLOSE EXPENSES FO
   R ";DA$
7920  PRINT D$"NOMON C,I,O"
7940 S1% = 1
7960  IF S5% = 1 THEN  RETURN
7980  GOTO 8660
8000  REM --------------------
8020  REM  CHANGE FILE NAME
8040  REM --------------------
8060  HOME :HOLD$ = DA$:S2% = 1:
   INVERSE :TI$ = "CHANGE DATE/
   NAME OF AN EXISTING REPORT":
   PRINT TI$: NORMAL : VTAB 4:
   PRINT "OPTIONS:": PRINT
   : PRINT
8080  PRINT "    1) CATALOG DRIVE
   "DR: PRINT : PRINT "    2)
   ENTER PRESENT DATE/NAME OF T
   HE           REPORT TO BE CHA
   NGED": PRINT : PRINT : PRINT
   "PRESS ESC TO RETURN TO THE
   MAIN MENU OR ENTER YOUR CHOI
   CE BY NUMBER PLEASE: ";: GET
   A$: PRINT A$
8100  IF A$ = CHR$ (27) GOTO 866
   0
8120  IF A$ = "1" THEN  HOME : PRINT
   D$"CATALOG, S"SL",D"DR: PRINT
   "PRESS ANY KEY TO CONTINUE:"
   ;: GET B$: PRINT B$
8140  IF A$ <  > "2" GOTO 8060
8160  HOME : INVERSE : PRINT TI$:
   NORMAL : PRINT : PRINT : PRINT
   "PLEASE ENTER THE CURRENT DA
   TE/NAME OF   THE REPORT WHIC
   H YOU WISH TO CHANGE."
8180 S3% = 0: GOSUB 7080
8200 DT$ = DA$
8220  HOME : INVERSE : PRINT TI$:
   NORMAL
8240  VTAB 6: PRINT "OLD FILE DAT
   E/NAME:": PRINT : PRINT "EXP
   ENSES FOR "DT$
8260  VTAB 20: PRINT "PLEASE ENTE
   R THE REPLACEMENT DATE/NAME:
   "
8280 S3% = 0: GOSUB 7080
8300  HOME : PRINT "PRESS ";: INVERSE
   : PRINT "RETURN";: NORMAL : PRINT
   " AND YOU WILL CHANGE:": PRINT
   : PRINT "EXPENSES FOR "DT$
8320  PRINT : PRINT "TO": PRINT :
   PRINT "EXPENSES FOR "DA$: PRINT
   : PRINT : PRINT : PRINT : PRINT
   "ANY OTHER KEY EXITS TO MENU
   -->";
8340  GET A$: PRINT A$: IF A$ <  >
   CHR$ (13) GOTO 8660
8360  PRINT D$"MON C,I,O"
8380 ERR$ = "ON"
8400  PRINT D$"RENAME EXPENSES FO
   R "DT$",EXPENSES FOR "DA$
8420 ERR$ = ""
8440  PRINT D$"NOMON C,I,O"
8460 DA$ = HOLD$: IF DA$ <  > "" THEN
   S3% = 1
8480  GOTO 8660
8500  REM --------------------
8520  REM ROUTINE TO CATALOG
8540  REM --------------------
8560  HOME : PRINT  CHR$ (13) CHR$
   (4)"CATALOG,S"SL",D"DR
8580  PRINT : PRINT "PRESS ANY KE
   Y TO CONTINUE. ": GET A$: PRINT
   A$: GOTO 8660
8600  REM --------------------
8620  REM **  MAIN MENU  **
8640  REM --------------------
8660  HOME : INVERSE : PRINT  SPC(
   14)"EXPENSE-CALC" SPC( 14): NORMAL
8680  VTAB 3: PRINT "OPTIONS: "
8700  VTAB 5: PRINT "1) INPUT A N
   EW REPORT"
8720  VTAB 7: PRINT "2) REVIEW/CH
   ANGE/DELETE/UPDATE"
8740  VTAB 9: PRINT "3) SEE CATAL
   OG OF THIS DISK'S FILES"
8760  VTAB 11: PRINT "4) PRINT HA
   RD COPY OF DATA"
8780  VTAB 13: PRINT "5) CHANGE D
   ATE/NAME OF A FILE"
8800  VTAB 15: HTAB 1: PRINT "6)
   CHANGE DRIVE # : DRIVE NOW =
   ";: INVERSE : PRINT DR: NORMAL
8820  VTAB 17: HTAB 1: PRINT "7)
   CHANGE SLOT # : SLOT NOW =
   ";: INVERSE : PRINT SL: NORMAL
8840  VTAB 19: PRINT "8) END"
8860 I$ = "PLEASE TYPE THE NUMBER
   OF YOUR CHOICE"
8880  GOSUB 3180: GET A$
8900  IF A$ =  CHR$ (13) THEN A$ =
   "X"
8920  PRINT A$;
8940  IF A$ = "1" THEN  HOME : PRINT
   OMP$: CLEAR :S% = 1:S2% = 1:
   S4% = 1:S5% = 1: GOSUB 11020
   :S2% = 0: GOTO 5380
8960  IF A$ = "2" GOTO 4100
8980  IF A$ = "3" GOTO 8560
```

```
9000  IF A$ = "4" THEN  GOSUB 922
   0: GOTO 8660
9020  IF A$ = "5" GOTO 8060
9040  IF A$ = "6" AND DR = 1 THEN
   DR = 2: POKE 769,DR: GOTO 88
   00
9060  IF A$ = "6" AND DR = 2 THEN
   DR = 1: POKE 769,DR: GOTO 88
   00
9080  IF A$ = "7" THEN  HOME : VTAB
   12: PRINT "WHICH SLOT # (1-7
   )? ";: GET SL$: IF SL$ < "1"
   OR SL$ > "7" GOTO 9080
9100  IF A$ = "7" THEN SL =  VAL
   (SL$): POKE 768,SL: GOTO 866
   0
9120  IF A$ = "8" THEN  HOME : END
9140  GOTO 8880
9160  REM -------------------
9180  REM PRINTER ROUTINE
9200  REM -------------------
9220  IF S3% = 0 THEN I$ = "ASSIG
   N A DATE/NAME TO THIS REPORT
   ": HTAB 1: VTAB 24: CALL  -
   868: GOSUB 3180: GOSUB 7140
9230  IF  LEFT$ (DA$,4) = "YEAR" THEN
   DI = IT(8,19) - HOLD(8,20)
9240  I$ =  CHR$ (9)
9250  DI = IT(8,19) - AD: IF  LEFT$
   (DA$,4) = "YEAR" THEN DI = I
   T(8,19) - HOLD(8,20)
9260  PR# 1
9280  PRINT I$"80N"
9300  PRINT : PRINT : PRINT  TAB(
   27)"EXPENSE REPORT FOR ";DA$
   : PRINT : PRINT
9320  PRINT "    ITEM     ";: FOR
   I = 1 TO 7: PRINT "  ";DA$(I
   )"  ";: NEXT : PRINT "   T
   OTAL":
9340  PRINT "-------------";: FOR
   I = 1 TO 7: PRINT "--------"
   ;: NEXT : PRINT "  -------"
9360  PRINT
9380  FOR EX = 1 TO 16
9400  PRINT EX$(EX);"  ";
9420  FOR DA = 1 TO 7
9440 A = IT(DA,EX): GOSUB 9800
9460  PRINT  TAB( 9 -  LEN (A$))A
   $;
9480  NEXT DA
9500 A = IT(8,EX): GOSUB 9800
9520  PRINT  TAB( 10 -  LEN (A$))
   A$
9540  NEXT EX
9560  FOR I = 1 TO 78: PRINT "-";
   : NEXT : PRINT "-"
9580  FOR EX = 17 TO 18
9600  PRINT EX$(EX);"  ";
9620  FOR DA = 1 TO 7:A = IT(DA,E
   X): GOSUB 9800: PRINT  TAB(
   9 -  LEN (A$))A$;: NEXT
9640 A = IT(8,EX): GOSUB 9800: PRINT
   TAB( 10 -  LEN (A$))A$
9660  NEXT EX
9680  FOR I = 1 TO 78: PRINT "-";
   : NEXT : PRINT "-"
9700  PRINT EX$(19);"  ";: FOR DA
   = 1 TO 7:A = IT(DA,19): GOSUB
   9800: PRINT  TAB( 9 -  LEN (
   A$))A$;: NEXT :A = IT(8,19):
   GOSUB 9800: PRINT  TAB( 10 -
   LEN (A$))A$
9710 A = ADV: IF  LEFT$ (DA$,4) =
   "YEAR" THEN A = HOLD(8,20)
9720  PRINT EX$(20);: GOSUB 9800:
   PRINT  TAB( 68 -  LEN (A$))
   A$
9740  PRINT EX$(21);:A = DI: GOSUB
   9800: PRINT  TAB( 68 -  LEN
   (A$))A$
9760  PRINT : PRINT : PRINT : PRINT
9780  PR# 0: RETURN
9800  IF A = 0 THEN A$ = "-": RETURN
9820 B$ =  STR$ ( INT ((A + .005)
   * 100)): IF  LEN (B$) > 2 THEN
   A$ =  LEFT$ (B$, LEN (B$) -
   2) + "." +  RIGHT$ (B$,2): RETURN
9840  IF  LEN (B$) = 2 THEN A$ =
   "." + B$: RETURN
9860 A$ = ".0" + B$: RETURN
9880  REM --------------------
9900  REM ONERR PRINT OUT AND
9920  REM ERR HANDLING ROUTINE
9940  REM --------------------
9960 X =  PEEK (222): REM  GET ER
   ROR CODE
9980  REM ** APPLESOFT ERRORS **
10000  IF X = 0 THEN  PRINT "NEXT
   WITHOUT FOR";
10020  IF X = 16 THEN  PRINT "SYN
   TAX";
10040  IF X = 22 THEN  PRINT "RET
   URN WITHOUT GOSUB";
10060  IF X = 42 THEN  PRINT "OUT
   OF DATA";
10080  IF X = 53 THEN  PRINT "ILL
   EGAL QUANTITY";
10100  IF X = 69 THEN  PRINT "OVE
   RFLOW";
10120  IF X = 77 THEN  PRINT "OUT
   OF MEMORY";
10140  IF X = 90 THEN  PRINT "UND
   EFINED STATEMENT";
10160  IF X = 107 THEN  PRINT "BA
   D SUBSCRIPT";
10180  IF X = 120 THEN  PRINT "RE
   DIMENSIONED ARRAY";
10200  IF X = 133 THEN  PRINT "DI
   VISION BY ZERO";
10220  IF X = 163 THEN  PRINT "TY
   PE MISMATCH";
10240  IF X = 176 THEN  PRINT "ST
   RING TOO LONG";
```

```
10260  IF X = 191 THEN  PRINT "FO
    RMULA TOO COMPLEX";
10280  IF X = 224 THEN  PRINT "UN
    DEFINED FUNCTION";
10300  IF X = 254 THEN  PRINT "BA
    D RESPONSE TO INPUT STATEMEN
    T";
10320  IF X = 255 THEN  PRINT "CT
    RL C INTERRUPT";
10340  REM   ****DOS MESSAGES***
10360  IF X = 1 THEN  PRINT "LANG
    UAGE NOT AVAILABLE";
10380  IF X = 2 OR X = 3 THEN  PRINT
    "RANGE ERROR";
10400  IF X = 4 THEN  PRINT "WRIT
    E PROTECTED";
10420  IF X = 5 GOTO 10820
10440  IF X = 6 AND ERR$ = "ON" THEN
    ERR$ = "OFF": GOTO 10820
10460  IF X = 6 GOTO 10720
10480  IF X = 7 THEN  PRINT "VOL.
     MISMATCH";
10500  IF X = 8 THEN  PRINT "I/O
    ERROR";
10520  IF X = 9 THEN  PRINT "DISK
     FULL";
10540  IF X = 10 THEN  PRINT "FIL
    E LOCKED";
10560  IF X = 11 THEN  PRINT "SYN
    TAX";
10580  IF X = 12 THEN  PRINT "NO
    BUFFERS AVAILABLE";
10600  IF X = 13 THEN  PRINT "FIL
    E TYPE MISMATCH";
10620  IF X = 14 THEN  PRINT "PRO
    GRAM TOO LARGE";
10640  IF X = 15 THEN  PRINT "NOT
     DIRECT COMMAND";
10660  PRINT " ERROR IN LINE # ";
    ( PEEK (218) + ( PEEK (219) *
    256))
10680  END
10700  REM CREATE NEW HEADER
10720  HOME : PRINT "THERE IS NO
    PREVIOUSLY ESTABLISHED
    EXPENSE HEADER FILE ON THIS
    DISK.": PRINT : PRINT "DO YO
    U WISH TO INITILIZE A HEADER
     ON    THIS DISK FOR A NEW E
    XPENSE PERIOD?    (Y OR N):
     ";: GET A$: PRINT A$
10740  IF A$ = "N" GOTO 8660
10760  IF A$ <  > "Y" GOTO 10720
10780  GOTO 6460
10800  REM    HANDLE REQUEST FOR
    BOGUS FILE
10820  TEXT : PRINT : HOME
10840  PRINT D$"OPEN EXPENSES FOR
    "DA$",L9,S"SL",D"DR
10860  PRINT D$"DELETE EXPENSES F
    OR "DA$",S"SL",D"DR
10880  PRINT : PRINT "THE DATE ";
    DA$;" IS NOT CURRENTLY": PRINT
    "IN MEMORY AS A DATE BEGININ
    G AN ESTAB-  LISHED EXPENSE
    PERIOD.": PRINT : PRINT "PRE
    SS ANY KEY TO CONTINUE: ": GET
    A$: PRINT A$
10900  PRINT D$"NOMON C,I,O"
10920 S3% = 0: GOTO 8660
10940  REM -------------------
10960  REM   DEFINE VARIABLES
10980  REM -------------------
11000  CLEAR
11020  ONERR  GOTO 9960
11040 OMP$ = "ONE MOMENT PLEASE"
11060  DIM IT(8,21)
11080  DIM DA$(8)
11100  DIM EX$(21)
11120  DIM HOLD(8,20)
11140 F$ = "FRMT,X6,S,2,0;"
11160 F2$ = "FRMT,X7,S,2,0;"
11180 F3$ = "FRMT,'$',X8,S,2,0;"
11200 INST$ = "RET(ENTER):ESC(^):
    <-:->:L(EFT:R(T:Q(UIT"
11220 DA$(4) = "WED"
11240 ADVANCE =  0 : REM  CUSTOM-
    IZE TO SUIT YOURSELF
11260 DA$(1) = "SUN"
11280 DA$(2) = "MON"
11300 DA$(3) = "TUE"
11320 DA$(4) = "WED"
11340 DA$(5) = "THR"
11360 DA$(6) = "FRI"
11380 DA$(7) = "SAT"
11400 DA$(8) = "TOT "
11420 OV$ = "OVRFLW"
11440 EX$(1) = " 1 BOOK/MAG"
11460 EX$(2) = " 2 BREAKFST"
11480 EX$(3) = " 3 LUNCH   "
11500 EX$(4) = " 4 DINNER  "
11520 EX$(5) = " 5 CINEMA  "
11540 EX$(6) = " 6 CLOTHES "
11560 EX$(7) = " 7 CIGRETTE"
11580 EX$(8) = " 8 FOOD    "
11600 EX$(9) = " 9 GAS     "
11620 EX$(10) = "10 WAT/ELEC"
11640 EX$(11) = "11 MORTAGE "
11660 EX$(12) = "12 PHONE   "
11680 EX$(13) = "13 SCL FREE"
11700 EX$(14) = "14 TRANSPRT"
11720 EX$(15) = "15 HOBBY   "
11740 EX$(16) = "16 OTHER   "
11760 EX$(17) = "SUBTOTAL ->"
11780 EX$(18) = "- CO CHGES."
11800 EX$(19) = "NET TOTAL->"
11820 EX$(20) = "ADVANCE    "
11840 EX$(21) = "DIFFERENCE "
11860 SL =  PEEK (768):DR =  PEEK
    (769)
11880 D$ =  CHR$ (4)
11900  FOR I = 1 TO 6:UNDERLINE$ =
    UNDERLINE$ +  CHR$ (95): NEXT
11920  IF S2% = 1 AND S% = 1 THEN
    RETURN
11940  IF S% = 1 GOTO 8660
11960  GOTO 3500
```

```
*** AMPER-INTERPRETER ***
*91A0.9607

91A0- A0 02 B9 B8 91 99 F5 03
91A8- 88 10 F7 A9 BB 85 73 85
91B0- 6F A9 91 85 74 85 70 60
91B8- 4C BB 91 A2 00 A0 00 84
91C0- 3A B1 B8 85 1C BD 05 92
91C8- F0 1C C5 1C D0 04 C8 E8
91D0- D0 EF 29 7F C5 1C F0 13
91D8- E6 3A A0 00 A9 80 3D 05
91E0- 92 D0 EC E8 D0 F6 A2 10
91E8- 4C 12 D4 18 C8 98 65 B8
91F0- 85 B8 90 02 E6 B9 06 3A
91F8- A6 3A E8 BD 3B 92 48 CA
9200- BD 3B 92 48 60 50 52 4E
9208- D4 43 4C 52 4C 49 CE 43
9210- 4C 52 50 41 C7 48 49 53
9218- 43 52 45 45 4E B2 48 49
9220- 53 43 52 45 45 CE 42 45
9228- 4C CC 43 4C 52 45 52 D2
9230- 45 52 50 54 43 C8 52 45
9238- 50 D4 00 4C 92 9B FC 41
9240- FC B0 95 A5 95 DC FB C9
9248- 95 CE 95 D8 95 20 BE DE
9250- A9 20 85 19 A9 00 85 1A
9258- 85 3D 85 3E 85 E3 20 7B
9260- DD A5 11 85 3C A2 A1 A0
9268- 95 20 2B EB 20 BE DE 20
9270- E3 DF 18 69 01 D0 01 C8
9278- 85 1C 84 1D A0 00 B1 1C
9280- 48 C8 B1 1C 85 1D 68 85
9288- 1C 88 B1 1C F0 05 D9 73
9290- 93 F0 05 A2 10 4C 12 D4
9298- C8 C0 05 D0 ED A2 00 86
92A0- 3B BD 78 93 F0 ED D1 1C
92A8- F0 03 E8 D0 F4 8A 0A AA
92B0- E8 BD 80 93 48 CA BD 80
92B8- 93 48 8A 38 E9 04 B0 03
92C0- 4C 72 93 F0 FB 4A C9 03
92C8- D0 02 A9 04 85 3A A5 B8
92D0- 85 26 A5 B9 85 27 18 98
92D8- 65 1C 85 B8 A9 00 65 1D
92E0- 85 B9 A9 00 85 1C A5 1C
92E8- 29 01 D0 0D 20 B1 00 20
92F0- 4A EC 20 FB E6 8A 4C 07
92F8- 93 20 B1 00 C9 53 D0 02
9300- A9 20 48 20 B1 00 68 A4
9308- 1C 99 18 00 C8 84 1C C4
9310- 3A D0 D3 A5 18 C5 1A B0
9318- 03 4C 93 92 A9 A1 A0 95
9320- 20 F9 EA A5 3C D0 2E 20
9328- 34 ED 85 1C 84 1D A0 00
9330- 84 1F B1 1C F0 14 C9 45
9338- D0 05 68 68 4C FC 94 C9
9340- 2E D0 04 84 1F E6 1F C8
9348- D0 E8 84 1E A5 1F D0 02
9350- 84 1F 4C 65 93 A0 00 B1
9358- A0 85 1E C8 B1 A0 85 1C
9360- C8 B1 A0 85 1D A2 00 86
9368- 3B 20 B7 00 C9 3A D0 02
9370- E6 3B 60 46 52 4D 54 2C
9378- 27 48 42 24 46 58 43 00
9380- 8D 93 AF 93 B9 93 D4 93
9388- 59 94 94 94 D5 94 C8 B1
9390- 1C D0 05 A2 10 4C 12 D4
9398- C9 27 F0 06 20 5C DB C8
93A0- D0 ED C8 B1 1C F0 08 C9
93A8- 2C D0 E8 C8 4C 9D 92 60
93B0- 20 2E 95 20 DA FD 20 45
93B8- 95 60 20 2E 95 85 3A A2
93C0- 08 06 3A 90 04 A9 31 D0
93C8- 02 A9 30 20 5C DB CA D0
93D0- F0 20 45 95 60 A5 3C D0
93D8- 03 4C 32 95 38 A5 1E F0
93E0- 6D E5 18 B0 39 49 FF A8
93E8- C8 A5 19 C9 30 D0 15 A6
93F0- 3D 10 11 E6 1C D0 02 E6
93F8- 1D A9 2D 20 5C DB C6 1E
9400- C6 3A A5 19 A6 3E F0 0E
9408- 38 98 E5 3E F0 14 B0 03
9410- 4C FC 94 A8 A5 19 20 5C
9418- DB 88 D0 FA F0 04 A5 18
9420- 85 1E A0 00 A5 3E F0 1C
9428- B1 1C C9 2E D0 06 A9 00
9430- 85 3E F0 10 C6 3A 10 0C
9438- A9 03 85 3A A9 2C 20 5C
9440- DB 4C 34 94 B1 1C 20 5C
9448- DB C8 C4 1E D0 D6 20 45
9450- 95 A5 26 85 B8 A5 27 85
9458- B9 60 A6 1E E4 18 F0 26
9460- 90 24 A6 1F E4 1E F0 01
9468- CA E4 18 F0 05 90 03 4C
9470- FC 94 E8 86 1F 38 A5 18
9478- E5 1F B0 02 69 01 85 1A
9480- 20 4D 95 4C 5A 94 A0 00
9488- B1 1C C9 2D D0 04 C6 3D
9490- E6 3A 4C DC 93 38 A5 1E
9498- E5 1F 38 E5 1A 90 0C F0
94A0- 29 20 4D 95 A5 3E D0 2E
94A8- 4C 95 94 49 FF AA E8 A4
94B0- 1E C4 1F D0 06 E8 A9 2E
94B8- 91 1C C8 18 8A 65 1E 85
94C0- 1E A5 1B 91 1C C8 C4 1E
94C8- D0 F9 A5 18 C5 1E B0 03
94D0- 4C FC 94 4C 86 94 A0 00
94D8- 84 3A 84 3E A6 1F E4 1E
94E0- F0 01 CA B1 1C C9 2D D0
94E8- 01 CA 8A C9 03 F0 08 90
94F0- 06 E9 03 E6 3E D0 F4 85
94F8- 3A 4C 95 94 A9 2A 85 B8
9500- A9 95 85 B9 20 E3 DF 85
9508- 1C 84 1D A0 00 84 3E B1
9510- 1C 85 1E C8 B1 1C 48 C8
9518- B1 1C 85 1D 68 85 1C A9
9520- 20 85 19 A9 01 85 E3 4C
9528- DC 93 4F 56 24 3A A5 3C
9530- F0 05 A2 A3 4C 12 D4 C8
9538- B1 1C C9 3A D0 02 E6 3B
9540- 20 FB E6 8A 60 A5 3B F0
9548- 03 20 FB DA 60 A9 64 A0
9550- EE 20 F9 EA A5 1A 85 3A
9558- A5 3A F0 07 20 55 EA C6
9560- 3A 10 F5 A9 A1 A0 95 AE
9568- A2 95 86 A2 20 BE E7 20
9570- 27 93 A5 1F C5 1E F0 28
9578- 18 65 1A C5 1E B0 02 85
9580- 1E A4 1E 88 B1 1C C9 2E
9588- D0 16 C6 1E C6 1F F0 07
9590- 88 B1 1C C9 2D D0 09 A0
9598- 00 A9 30 91 1C C8 84 1E
95A0- 60 84 00 00 00 00 2C 54
95A8- C0 2C 53 C0 A9 20 4C B9
95B0- 95 2C 55 C0 2C 52 C0 A9
95B8- 40 85 E6 2C 57 C0 2C 50
95C0- C0 A9 01 85 E7 A9 00 85
95C8- F9 60 A9 00 85 D8 60 68
95D0- A8 68 A6 DF 9A 48 98 48
95D8- 60 20 BE DE 20 7B DD 20
95E0- FB E6 8A 48 20 BE DE 20
95E8- 7B DD 20 FB E6 68 20 5C
95F0- DB CA D0 FA 60 D0 FA 60
95F8- 35 30 30 30 0D 0D 0D 01
9600- 1C C8 84 1E 60 84 00 00
```

比RAM DISK使用更加方便

RAM咭控制中心

蘋果長老

大型電腦的其中之一的優點是可以完全不影响機內記憶而執行一系列的系統程式，例如EDITOR，RENUMBER或AUTO－LINE NUMBERING等，但APPLE上却沒有這些優點，這是我編寫本程式的原因。

本程式可以將十個機械語言程式或APPLESOFT 程式組合爲一個檔案，當這個檔案被BRUN時，會將原來的程式載入16K RAM 咭之上，而它們可以用一個& 來呼叫，更好的是因爲程式是貯存在RAM咭之上直到被呼叫爲止，它們不會浪費任何主機板上的記憶。

RAM SYSTEM 程式是由兩個檔案構成① RAM SYSTEM及②RAM SYSTEMB。在使用本程式時只需按RUN RAM SYSTEM 即可。這時會出現主目錄表（圖1），它可以讓你選擇4項功能，

1 選擇程式以載入RAM咭
2 刪除RAM咭上的程式
3 將RAM咭上的程式貯存到磁碟上
4 中止本程式

以下我將會解釋各項功能的使用。

第一項功能是讓你選擇磁碟上的程式來載入RAM咭，這些程式可以多達十個，唯它們的檔型必須爲A或B型，它們可以是你自己喜愛的程式，它們會被放到一個WORKFILE 之上來作暫存，之後會被合併成爲一個整體性的檔案，不過各位在選擇程式之時，要考慮一下它們的總長度不能大於12287個BYTES，因爲這是16K RAM咭上的程式存放區的空間，另4K要保留作爲指標向量表之用。目錄上的FREE BYTES亦即餘下的空間，大家要留意它。

選擇了這個功能之時程式會問你SLOT及 DRIVE號碼，這是你希望取出的程式 存放在那一架磁碟機上的號碼，若是沒有錯的話可以按RETURN接受自設值，之後會問你FILE TO SELECT：並給予你一行空行來鍵入程式名字，完成後按RETURN即會將程式載入。

選擇第二項功能可以將你WORKFILE 內的檔案刪除，你只要按入要刪除的程式號碼並按RETURN即可。

選擇第三項功能是將WORKFILE 內的程式們合併而成爲一個程式並貯到磁碟上。

程式會問你程式貯回磁碟上採用的檔名及AMPERSAND & ADDRESS，自設值爲$3CA，然後問目錄MENU存放的位置，長度將會是$2E9，因此不要大於$9317以免破壞DOS。

第4項功能可以離開本程式。

當我們完成將WORKFILE 貯回磁碟上之後，以後當你要取用之時只需以BRUN命令執行即可，該程式會自動移到SLOT 0上的RAM咭之中，使用時你只需按“&”及RETURN鍵，螢幕上會出現一些程式名稱，你把該程式的號碼按入後RETURN，程式便會自動由RAM咭移往正確的記憶區內執行。

```
COMMAND < >       1. SELECT A PROGAM
                  2. DESELECT A PROGAM
FREE BYTES        3. SAVE A UTILITY FILE
 <12287>          4. QUIT
                  5. CATALOG
CURRENT FILES:              SLOT <6>
                            DRIVE <1>
  TYPE  LENGTH  FILE
```

圖(一)

BOOT機

假如我們在中途重新BOOT機，在RAM咭上的RAM SYSTEM仍然會保留，唯“&”網路會失去，要重新連結你可以重新BRUN它，也可以將以下的一段程式作爲HELLO程式，它會自動檢查是否只要連結&指標或將檔案重新BRUN。（列表1）

表(一)　RAM BOOTING UP

```
1000  FOR I = 0 TO 24: READ A: POKE
     768 + T,A: NEXT T
1010  CALL 769: IF  PEEK (768) >
     0 THEN 1040
1020  POKE 1013,76: POKE 1014,202
     : POKE 1015,3: FOR T = 0 TO
     5: READ A: POKE 970 + T,A: NEXT
     T
1030  PRINT : PRINT "RAM SYSTEM E
     NABLE,AMPERSAND RESET": END
1040 F$ = "MYFILE": REM  F$=UTILI
     TY FILE TO BE LOADED
1050  PRINT : PRINT "LOADING UTIL
     ITY INTO LANGUAGE CARD": PRINT
      CHR$ (4);"BRUN ";F$
1060  DATA  0,173,136,192 ,172,1,
     208,174,2,208,173,138,192,19
     2,39,208,4,224,154,240,3,238
     ,0,3,96,173,136,192,76,3,208
```

程式的改良

RAM SYSTEM 雖然可以讓記憶體同時存放多組程式，不過它會佔用到＄9317 以上的空間，這樣我們在執行一些很長的程式時便會不大可能，而且我們若要將DOS移上RAM咭以增加10K使用空間便更不可能了，爲此我希望利用第二塊16KRAM咭來作這種用途，目前，我正在研究中。當然，我亦會考慮到將這程式應用在128K RAM 咭上，讀者們請留心日後發展。同時，我亦希望讀者之中，對16K RAM應用熟悉的跟我一齊思考使用第二張16K R—AM咭的可能性，假如有結果可電電腦時代出版社編輯部代傳達。

本程式在程式磁碟上的FILENAME是：

1. RAM SYSTEM
2. RAM SYSTEM B
3. RAM BOOTING UP

```
10   REM  ***********************
11   REM  *     RAM-SYSTEM      *
12   REM  *                     *
13   REM  * COPYRIGHT (C) 1984 *
14   REM  * BY COMPUTING AGE   *
15   REM  *                     *
16   REM  ***********************
100  HIMEM: 24063: PRINT  CHR$ (4
     );"BLOAD RAM-SYSTEM.B": POKE
     24657,94: POKE 24658,99: DIM
     C$(5),F$(9),M(9,2),TY$(9)
140 A = 24584:AA = 24435:C = 256:
     CD = 24434:D = 1:FS = 12287:
     I = 24410:M = 24676:P = 2465
     7:PS = 25438:M(0,1) =  PEEK
     (P) +  PEEK (P + 1) * C + 1:
     S = 6:TF =  - 1:B$ =  CHR$ (
     8):H$ = "0123456789ABCDEF":R
     $ =  CHR$ (13):U$ =  CHR$ (9
     5): FOR T = 1 TO 5: READ C$(
     T): NEXT T: FOR T = 0 TO 28:
      READ T1: POKE 768 + T,T1: NEXT
     T
170  TEXT : HOME : PRINT "COMMAND
      < >";
180  FOR T = 1 TO 5: HTAB 15: PRINT
     T;". ";C$(T): NEXT T: VTAB 3
     : PRINT "FREE BYTES": PRINT
     " <";FS;">": PRINT : PRINT "
     CURRENT FILES:"; TAB( 28);"S
     LOT  <";S;">": PRINT  TAB( 2
     8);"DRIVE <";D;">": PRINT  TAB(
     3);"TYPE  LENGTH  FILE": VTAB
     22: PRINT "** COPYRIGHT 1984
     COMPUTING AGE LTD  **": VTAB
     9:L = 12 + TF: IF TF < 0 THEN
     240
220  FOR T = 0 TO TF: PRINT T;"."
     ; TAB( 5);TY$(T); TAB( 9);:F
     L = M(T,2) - M(T,1) + 1: IF
     FL < 10000 THEN  FOR T1 = 1 TO
     (5 -  LEN ( STR$ (FL))): PRINT
     " ";: NEXT T1
230  PRINT FL; TAB( 17); LEFT$ (F
     $(T),24);: NEXT T: PRINT
240  VTAB 1: HTAB 10: GET T$:T =
      VAL ( LEFT$ (T$,1)): IF T <
     1 OR T > 5 THEN 180
250  PRINT T$: ON T GOSUB 310,530
     ,700,940,1155: GOTO 170
310  IF FS < 35 THEN  VTAB L: PRINT
     "THERE IS NO ROOM TO ADD ANO
     THER FILE.": GOSUB 1150: RETURN
320  IF TF = 9 THEN  VTAB L: PRINT
     "ONLY 10 FILES CAN BE SELECT
     ED AT A TIME": GOSUB 1150: RETURN
330  GOSUB 1000:F$ = "FILE TO SEL
     ECT:": GOSUB 1080: IF  LEFT$
     (F$,1) =  CHR$ (32) THEN  RETURN
360 T =  PEEK (P) +  PEEK (P + 1)
      * C: FOR T1 = 1 TO 30: POKE
     T + T1, ASC ( MID$ (F$,T1,1)
     ) + 128: NEXT T1: POKE I,S *
     16: POKE I + 1,D:L = L + 2: VTAB
     L: PRINT "INSERT DISK: <CR>
     CONTINUES <SP> ABORTS";
390  GET T$: IF T$ <  > R$ AND T$
      <  >  CHR$ (32) THEN 390
400  IF T$ =  CHR$ (32) THEN  RETURN
410  CALL 24064:T1 =  PEEK (CD): IF
     T1 < 255 THEN 460
440  POKE T + 32,T1:TF = TF + 1:F
     $(TF) = F$:M(TF,2) =  PEEK (
     P) +  PEEK (P + 1) * C: IF T
     F < 9 THEN M(TF + 1,1) = M(T
     F,2) + 1
450 FS = FS - (M(TF,2) - M(TF,1) +
     1):TY$(TF) =  CHR$ (65 + ( PEEK
     (M(TF,1) + 30) = 34)): RETURN
460  IF T1 = 0 THEN  VTAB L: HTAB
     1: PRINT "THE FILE ";F$: CALL
      - 868: PRINT "IS NOT ON THI
     S DISK.": GOSUB 1150: RETURN
470  IF T1 = 1 THEN  VTAB L: HTAB
     1: PRINT "THE FILE ";F$: CALL
      - 868: PRINT "IS NOT AN A O
     R B TYPE FILE": GOSUB 1150: RETURN
480 T1 =  PEEK (T + 31):T1 =  PEEK
     (T + T1 + 1) +  PEEK (T + T1
      + 2) * C + T1 + 2: VTAB L: HTAB
```

```
   1: PRINT "THE FILE ";F$: CALL
    - 868: PRINT "IS TOO LONG,
   ";T1;" BYTES.": GOSUB 1150: RETURN
530  IF TF < 0 THEN  RETURN
540  VTAB L: PRINT "PROGRAM TO DE
   SELECT (0-";TF;"):  < >";B$;
   B$;
550  GET T$:T =  ASC (T$): IF (T <
   48 OR T > 48 + TF) AND T$ <
    > R$ THEN 550
560  PRINT T$:T = T - 48: IF T$ =
   R$ THEN  RETURN
570 L = 0: IF TF = 0 THEN 650
580  FOR T1 = 0 TO TF: IF T <  >
   T1 THEN L = L + (M(T1,2) - M
   (T1,1) + 1)
590  NEXT T1: IF T = TF THEN 650
600  POKE 769,M(T + 1,1) -  INT (
   M(T + 1,1) / C) * C: POKE 77
   3, INT (M(T + 1,1) / C): POKE
   777,M(TF,2) -  INT (M(TF,2) /
   C) * C: POKE 781, INT (M(TF,
   2) / C): POKE 785,M(T,1) -  INT
   (M(T,1) / C) * C: POKE 789, INT
   (M(T,1) / C): CALL 768
640  FOR T1 = T TO (TF - 1):M(T1,
   2) = M(T1,1) + (M(T1 + 1,2) -
   M(T1 + 1,1)):M(T1 + 1,1) = M
   (T1,2) + 1:TY$(T1) = TY$(T1 +
   1):F$(T1) = F$(T1 + 1): NEXT
   T1
650 TF = TF - 1:FS = 12287 - L:L =
   L + PS: POKE P,L -  INT (L /
   C) * C: POKE P + 1, INT (L /
   C): RETURN
700  IF TF < 0 THEN  RETURN
710  GOSUB 1000:F$ = "FILE TO SAV
   E PROGRAMS IN:": GOSUB 1080:
   FS$ = F$: IF  LEFT$ (FS$,1) =
    CHR$ (32) THEN  RETURN
730 L = L + 1:F$ = "AMPERSAND & A
   DDRESS:":T1$ = "3CA"
740  VTAB L: PRINT F$;" ($";T1$;"
   ) ";: CALL  - 868: INPUT T$:
   T = 970: GOSUB 1190: IF T <
   0 OR T > 49151 THEN 740
750  IF  LEN (T$) = 0 THEN T$ = T
   1$
760  VTAB L: PRINT F$;" <$";T$;">
   ";: CALL  - 868: PRINT : POKE
   A,T -  INT (T / C) * C: POKE
   A + 5, INT (T / C): FOR T1 =
   12 TO 37 STEP 5: POKE A + T1
   ,T -  INT (T / C) * C: POKE
   A + T1 + 1, INT (T / C):T =
   T + 1: NEXT T1:L = L + 1:F$ =
   "MENU ADDRESS:":T1$ = "9317"
800  VTAB L: PRINT F$;" ($";T1$;"
   ) ";: CALL  - 868: INPUT T$:
   T = 37655: GOSUB 1190: IF T <
   0 OR T > 37655 THEN 800
810  IF  LEN (T$) = 0 THEN T$ = T
   1$
820  VTAB L: PRINT F$;" <$";T$;">
   ";: CALL  - 868: PRINT : POKE
   M,T -  INT (T / C) * C: POKE
   M + 4, INT (T / C): POKE M +
   16,T -  INT (T / C) * C: POKE
   M + 17, INT (T / C):L = L +
   1: IF TF < 9 THEN L = L + 1
850  VTAB L: PRINT "INSERT DISK:
   <CR> CONTINUES <SP> ABORTS";
860  GET T$: IF T$ <  > R$ AND T$
    <  >  CHR$ (32) THEN 860
870  IF T$ =  CHR$ (32) THEN  RETURN
880  POKE M(TF,1) + 31,0:L = M(TF
   ,2) - AA + 1: PRINT : PRINT
    CHR$ (4);"BSAVE";FS$;",A";A
   A;",L";L;",S";S;",D";D: POKE
   M(TF,1) + 31,255: RETURN
940  VTAB L: INPUT "DO YOU REALLY
    WISH TO QUIT ?";T$:T$ =  MID$
   (T$,1,1): IF T$ <  > "Y" THEN
   RETURN
950  HIMEM: 38400: VTAB 22: END
1000  VTAB 6: HTAB 35: GET T$:T =
   VAL (T$): IF (T < 1 OR T >
   7) AND T$ <  > R$ THEN 1000
1010  IF T$ <  > R$ THEN S = T: PRINT
   T$
1020  VTAB 7: HTAB 35: GET T$:T =
   VAL (T$): IF (T < 1 OR T >
   2) AND T$ <  > R$ THEN 1020
1030  IF T$ <  > R$ THEN D = T: PRINT
   T$
1040  RETURN
1080  VTAB L: HTAB 1: PRINT F$: PRINT
   TAB( 5);"<";: FOR T = 1 TO
   30: PRINT U$;: NEXT T: PRINT
   ">":F$ = "":T = 0: VTAB L +
   1: HTAB 6: FOR T1 = 1 TO 100
   00: GET T$: IF T$ <  > R$ AND
   T$ <  > B$ AND T < 30 AND NOT
   (T = 0 AND (T$< "A" OR T$ >
   "Z")) THEN F$ = F$ + T$:T =
   T + 1: PRINT T$;: NEXT T1
1100  IF T$ = R$ THEN  VTAB L: HTAB
   1: CALL  - 868: PRINT "FILE:
    <";F$;">": CALL  - 868: FOR
   T = 1 TO 30:F$ = F$ + " ": NEXT
   T:F$ =  LEFT$ (F$,30): RETURN
1110  IF T$ <  > B$ AND T = 30 THEN
   NEXT T1
1120  IF T = 0 AND (T$ < "A" OR T
   $ > "Z") THEN  NEXT T1
1130  IF T > 0 THEN  PRINT B$;U$;
   B$;:T = T - 1:F$ =  MID$ (F$
   ,1,T)
1140  NEXT T1
1150  VTAB 24: HTAB 8: PRINT "HIT
```

```
        ANY KEY TO CONTINUE ";: GET
     T$: RETURN
1155  GOSUB 1000: HOME :D$ =  CHR$
     (4): PRINT : PRINT D$;"CATAL
     OG,S"S",D"D: GOSUB 1150: RETURN

1190  IF  LEN (T$) > 4 THEN T =  -
     1: RETURN
1200  IF  LEN (T$) = 0 THEN  RETURN

1210 T = 0: FOR T1 = 1 TO  LEN (T
     $): FOR T2 = 0 TO 15: IF  MID$
     (T$, LEN (T$) + 1 - T1,1) <
     >  MID$ (H$,T2 + 1,1) THEN
     NEXT T2:T =  - 1: RETURN
1220 T = T + T2 * 16 ^ (T1 - 1): NEXT
     T1: RETURN
1230  DATA  SELECT A PROGAM,DESEL
     ECT A PROGAM,SAVE A UTILITY
     FILE,QUIT,CATALOG: DATA 169,
     0,133,60,169,0,133,61,169,0,
     133,62,169,0,133,63,169,0,13
     3,66,169,0,133,67,160,0,76,4
     4,254
```

RAM SYSTEM B

```
5E00- AD 51 60 85 0A AD 52 60
5E08- 85 0B 20 4E 5F A9 11 8D
5E10- 5D 5F A9 00 8D 5E 5F A9
5E18- 00 8D 61 5F A9 95 8D 62
5E20- 5F 20 46 5F AD 01 95 F0
5E28- 31 8D 5D 5F AD 02 95 8D
5E30- 5E 5F 20 46 5F A9 2B 48
5E38- AA A0 1E BD 00 95 D1 0A
5E40- D0 0D CA 88 D0 F5 CA CA
5E48- BD 00 95 C9 FF D0 0F 68
5E50- C9 FD F0 D0 18 69 23 4C
5E58- 37 5E 8D 72 5F 60 68 A0
5E60- 1F E8 E8 BD 00 95 29 0F
5E68- 48 18 69 1E 91 0A C9 20
5E70- F0 0E C9 22 F0 0A A9 01
5E78- 8D 72 5F 68 8D 6E 5F 60
5E80- CA BD 00 95 8D 5E 5F CA
5E88- BD 00 95 8D 5D 5F 20 46
5E90- 5F 18 A5 0A 69 21 8D 61
5E98- 5F A5 0B 69 00 8D 62 5F
5EA0- A2 0C BD 00 95 8D 5D 5F
5EA8- E8 BD 00 95 8D 5E 5F 20
5EB0- 46 5F A0 1F B1 0A 48 A8
5EB8- C8 B1 0A AA C8 B1 0A A8
5EC0- 8A 18 65 0A 85 0A 98 65
5EC8- 0B 85 0B 18 68 69 02 65
5ED0- 0A 85 0A 90 02 E6 0B A5
5ED8- 0B C9 93 90 0F D0 06 A5
5EE0- 0A C9 5E 90 07 A9 02 8D
5EE8- 72 5F 68 60 8E 6E 5F 8C
5EF0- 6F 5F 68 AA A9 00 8D 70
5EF8- 5F CE 70 5F CA D0 FA 8D
5F00- 71 5F A9 0D 48 AD 71 5F
5F08- CD 6F 5F 90 0A D0 26 AD
5F10- 70 5F CD 6E 5F B0 1E 68
5F18- AA E8 BD 00 95 8D 5D 5F
5F20- E8 BD 00 95 8D 5E 5F 8A
5F28- 48 EE 62 5F 20 46 5F EE
5F30- 71 5F 4C 05 5F A5 0A 8D
5F38- 51 60 A5 0B 8D 52 60 A9
5F40- FF 8D 72 5F 68 60 A9 5F
5F48- A0 59 20 D9 03 60 A9 6A
5F50- 8D 5F 5F A9 5F 8D 60 5F
5F58- 60 01 60 01 00 12 06 00
5F60- 00 00 00 00 00 01 00 00
5F68- 60 01 00 01 EF D8 85 A0
5F70- A0 C2 89 AD 89 C0 AD 89
5F78- C0 A9 27 8D 01 D0 A9 9A
5F80- 8D 02 D0 AD 8A C0 AD 88
5F88- C0 AD 01 D0 C9 27 D0 07
5F90- AD 02 D0 C9 9A F0 1A AD
5F98- 8A C0 20 8E FD 20 8E FD
5FA0- A2 1E BD 32 60 20 ED FD
5FA8- CA 10 F7 20 8E FD 4C D0
5FB0- 03 AD 8A C0 AD 89 C0 AD
5FB8- 89 C0 A9 53 85 3C A9 60
5FC0- 85 3D A9 5F 85 3E A9 63
5FC8- 85 3F A9 03 85 42 A9 D0
5FD0- 85 43 A0 00 20 2C FE AD
5FD8- 8A C0 AD 81 C0 AD 81 C0
5FE0- A9 5F 85 3C A9 63 85 3D
5FE8- AD 51 60 85 3E AD 52 60
5FF0- 85 3F A9 01 85 42 A9 D0
5FF8- 85 43 A0 00 20 2C FE AD
6000- 82 C0 A9 4C 8D F5 03 A9
6008- CA 8D F6 03 A9 03 8D F7
6010- 03 A9 AD 8D CA 03 A9 88
6018- 8D CB 03 A9 C0 8D CC 03
6020- A9 4C 8D CD 03 A9 03 8D
6028- CE 03 A9 D0 8D CF 03 4C
6030- D0 03 AE C4 C5 D4 D2 CF
6038- C2 C1 A0 C4 C1 CF CC A0
6040- AC C4 D2 C1 C3 A0 C5 C7
6048- C1 D5 C7 CE C1 CC A0 CF
6050- CE 83 B0 A9 26 85 3C A9
6058- D0 85 3D A9 0E 85 3E A9
6060- D3 85 3F A9 17 85 42 A9
6068- 93 85 43 A0 00 B8 20 10
6070- BA 50 7B 4C 17 93 AD 8A
6078- C0 AD 80 C0 AD 51 C0 AD
6080- 52 C0 AD 54 C0 A9 00 85
6088- 0A 85 25 85 47 85 48 A9
6090- 04 85 0B 85 49 AD 82 C0
6098- 20 58 FC AD 80 C0 A0 00
60A0- EA EA EA EA EA EA EA A9
60A8- D0 85 0B E6 0A A9 AF 48
60B0- 68 18 69 01 48 B8 20 10
60B8- BA 50 6A A9 AE B8 20 10
60C0- BA 50 62 A9 A0 B8 20 10
60C8- BA 50 5A A0 00 B1 0A B8
60D0- 20 10 BA 50 50 C8 C0 1E
60D8- D0 F3 B1 0A AA C8 B1 0A
60E0- 48 8A 18 65 0A 85 0A 90
60E8- 07 E6 0B 2C 10 BA 50 71
60F0- A0 00 B1 0A 48 C8 B1 0A
60F8- AA 68 18 65 0A 85 0A 8A
6100- 65 0B 85 0B A9 02 18 65
6108- 0A 85 0A 90 02 E6 0B A9
6110- 8D B8 20 10 BA 50 0E A9
6118- 8D B8 20 10 BA 50 06 68
6120- 30 8E 2C 10 BA 50 6E 20
6128- 10 BA BA CA 18 BD 00 01
6130- 69 19 85 3C E8 BD 00 01
6138- 69 02 85 3D A0 00 B1 3C
6140- B8 20 10 BA 50 4F C8 C0
6148- 09 D0 F3 68 85 0A A9 60
6150- B8 20 10 BA 50 3F AD 00
6158- C0 10 FB 2C 10 C0 2C 10
6160- BA 50 67 C9 8D F0 0C C9
6168- B0 90 EB C5 0A F0 10 90
6170- 0E B0 E3 C6 47 A9 A0 B8
6178- 20 10 BA 50 18 70 4D C6
6180- 47 B8 20 10 BA 50 0E 29
6188- 0F AA A9 01 85 0A A9 D0
6190- 85 0B 2C 10 BA 50 37 CA
6198- 30 26 A0 1E B1 0A 48 A8
61A0- B1 0A 48 C8 B1 0A A8 68
61A8- 18 65 0A 85 0A 98 65 0B
61B0- 85 0B 68 18 69 02 65 0A
61B8- 85 0A 90 DB E6 0B D0 D7
61C0- A0 1E B1 0A C9 22 F0 7C
61C8- D0 06 50 72 70 72 50 72
61D0- A8 B1 0A 38 E9 01 48 C8
61D8- B1 0A E9 00 48 18 A9 22
61E0- 65 0A 85 0A 85 3C A5 0B
61E8- 69 00 85 0B 85 3D 68 A8
61F0- 68 48 18 65 0A 85 3E 98
61F8- 48 65 0B 85 3F A5 67 85
6200- 42 A5 68 85 43 A0 00 B8
6208- 20 10 BA 50 31 68 A8 68
6210- 18 65 67 85 AF 85 69 98
6218- 65 68 85 B0 85 6A E6 69
6220- D0 02 E6 6A A6 68 A4 67
6228- D0 01 CA 88 84 0A 86 0B
6230- A0 00 98 91 0A AD 82 C0
6238- 20 58 FC 4C 66 D5 50 54
6240- 70 74 50 1F A0 20 B1 0A
6248- 85 42 85 47 C8 B1 0A 85
6250- 43 85 48 C8 B1 0A 38 E9
6258- 01 48 C8 B1 0A E9 00 48
6260- 2C 10 BA 50 7A 18 A9 24
6268- 65 0A 85 0A 85 3C A5 0B
6270- 69 00 85 0B 85 3D 68 A8
6278- 68 18 65 0A 85 3E 98 65
6280- 0B 85 3F A0 00 B8 20 10
6288- BA 50 09 AD 82 C0 20 58
6290- FC 6C 47 00 BA CA CA 9A
6298- B1 3C 91 42 E6 42 D0 02
62A0- E6 43 A5 3C C5 3E A5 3D
62A8- E5 3F E6 3C D0 02 E6 3D
62B0- 90 E6 2C 10 BA 60 A9 8D
62B8- B8 20 10 BA 50 21 A9 8D
62C0- B8 20 10 BA 50 19 20 10
62C8- BA BA BD 00 01 85 3D CA
62D0- BD 00 01 18 69 83 85 3C
62D8- 90 02 E6 3D 2C 10 BA 50
62E0- 1B A0 13 B1 3C B8 20 10
62E8- BA 50 11 88 10 F5 A9 8D
62F0- B8 20 10 BA 50 06 AD 82
62F8- C0 4C D0 03 86 42 BA CA
6300- CA 9A A6 42 C9 8D F0 0F
6308- 84 42 A4 47 91 48 C8 84
6310- 47 A4 42 2C 10 BA 60 A9
6318- 00 85 47 E6 25 18 A5 48
6320- 2A 48 A5 49 2A 69 01 6A
6328- 85 49 68 6A 85 48 A5 49
6330- C9 08 D0 0A 4A 85 49 A5
6338- 48 18 69 28 85 48 2C 10
6340- BA 60 D0 D2 CF C7 D2 C1
6348- CD A0 BF AE C4 C5 D4 C3
6350- C5 CC C5 D3 A0 CD C1 D2
6358- C7 CF D2 D0 A0 CF CE 80
6360- FF FF FF FF FF FF FF FF
```

PRODOS 縱橫

普魯斯

BASIC解譯器機械碼進入點

PRODOS本身是由很多個機械語言的副程式所組成，它們被置於 RAM 咭之內，而 BASIC.SYSTEM 是 BA—SIC 用家與 PRODOS之間的介面，在其中 BASIC 所發出的命令將被組繹執行，不過在 多方面的 角度下看 BASIC.SYSTEM ，其中的BI（BASIC INTERPRETER)與 DOS 3.3 中的有很多相似，而筆者今次的目的是將這個BI 反組繹希望得出其中處理各個命令的ML位置。

筆者花了多個小時來研究 BI 中的結構之後，加上從 PRODOS TEC—HNICAL REFERENCE MAN—UAL ，第 118—124 頁 BI GLOBAL PAGE的解說，終於發現了各個命令的進入點，它們到於表一之中，相信各位喜歡機械語言的朋友們都會喜歡知道。■

9A00	—9A16	配置I／O網路到 CSW／KSW 位置（使 PRODOS不作用）
9A17	—9A2E	重置 PRODOS（使PRODOS作用）
9A2F	—9A8C	截取影像輸出的網路
9A8D	9AA2	貯存眞正的 I／O網路來連結 PRODOS
9AA3	9AB9	將 PRODOS的攔截網路連接到 CSW／KSW去。
9ABA	9B1F	截取鍵盤網路
9B20	9BA4	承接 ERROR 網路
9FBF	A024	輸出 ERROR 訊息網路
A07B	A288	承接 FRE 命令
A6B4	A6D8	分隔命令行號
A6D9	A83C	處理命令
A83D	AA9B	執行命令
AA9C	AAED	將 ASC II十進制數字轉換爲三個 BYTE 的十六進制數字
AAEE	AB20	將 ASC II十六進制數字轉換爲三個 BYTE 的十六進制數字
AB21	AB82	檢查命令是否正確
AB83	ABCF	承接"—"命令
ABPO	ABF1	承接"CHAIN"命令
ABFZ	AC2F	承接"RUN" 命令
AC30	ACFF	承接"LOAD" 命令
AC35	AC3F	DOS 的 WARM START 網路
AD00	AD76	承接"SAVE"命令
AD77	ADB5	承接"CREATE"命令
ADB6	ADC1	承接"RENAME" 命令
ADC2	ADC6	承接"DELETE"命令
ADC7	ADD6	承接"LOCK" 命令
ADD7	ADE4	承接"UNLOCK"命令
ADE5	AE1C	承接"PREFIX" 命令
AE1D	AE5A	承接"BSAVE"命令
BA53	BA65	BI 錯誤表
BA66	BA7A	BI 初定值
BA7B	BA99	錯誤訊號表
BA9A	BB98	錯誤字位元表
BB99	BBB5	程式的 LOGIC TEXT
BCBD	BDFE	TEXT DATA 緩衝器

■

機械語言入門 張博士

加減乘除四則運算

今期我要和各位討論一下如何利用機械語言來作加減乘除四則運算，我希望能夠以深入淺出的方法使各位容易學會。

儘管ＢＡＳＩＣ在處理四則運算時十分簡易，可以說無需去學，但機械語言處理四則運算却不十分容易，可以說很麻煩（與ＢＡＳＩＣ比較），但我們可以將它廣泛應用到程式之中，取得ＢＡＳＩＣ無可比擬的高速運算。

一個 BYTE 的加法

首先加法運算的中心是累加器，6502 微處理器只有累加器才可以作加法運算，假如你希望去加上兩個數字並顯示出結果，我們要編寫出以下的程式：

```
LDA  $390     將第一個數載入累加器
CLC           清除進位位元
ADC  $391     將累加器的數字和$391位址內的數字
              相加
JSR  $FDDA    顯示結果
RTS           結束
```

以上的ADC命令會使用到狀態暫存器內的CARRY BIT（進位位元），假如我們要將＄90和＄80相加的話，結果應該是＄110的，不過累加器內却只有＄10，因爲數值大於＄FF（8 BIT 數最大値），因此要將爲首的1放到進位位元去，因此進位位元會設定爲1。

除了ADC這個命令以外，還有兩個命令會連帶進位位元，它們是BCC［BRANCH IF CARRY BIT IS CLEAR］及BCS［BRANCH IF CARRY BIT IS SET TO 1］，BRANCH（分歧）的意思即相當於我們的 GOTO 命令，以下的例子是應用以上兩個命令來作加法運算的。

```
CLC   ;          清除進位位元使之等於0
LDA  $390        將第一個數字載入累加器
ADC  $391        將第二個數字和累加器內第二個數
                 字相加
STA  $392        將加後結果貯存到$392位址上
BCC  DISPLAY     若果進位位元＝0則顯示結果
LDA  #$B1        若進位位元＝1，先在結果前印出
                 1
JSR  $FDED
—DISPLAY————
LDA  $392        將$392內的結果顯示
JSR  $FDDA
RTS
```

兩個 BYTE 的加法

要將兩個分別爲兩個ＢＹＴＥ的數字相加，首先要將它們的ＬＳＢ（LEASE SIGNIFICANT BYTE）相加（卽右邊的一個ＢＹＴＥ），若果有進位的話，進位位元會被設定爲1，而當ＭＳＢ被處理加法時，累加器會連同進位位元一起計算。

以下是一個使用例子：

```
JSR  $FC58    清除螢幕
LDA  #00    ⎫
LDA  $394   ⎬ 設定貯存運算結果
STA  $395   ⎪
STA  $396   ⎭
CLC           清除進位位元
LDA  $391   ⎫ 將兩個ＬＳＢ相加
ADC  $393   ⎭
STA  $396     將結果貯存在$396
LDA  $390   ⎫ 將ＭＳＢ相加，並連同進位位元相加
ADC  $392   ⎭
STA  $395     將結果貯存在$395
LDA  $394   ⎫ 計算進位位元
ADC  #$00   ⎭
STA  $394     貯回結果至$394
LDA  $394     顯示ＭＳＢ的數值
JSR  $FDDA
LDA  $395   ⎫
JSR  $FDDA  ⎬ 顯示其餘兩個ＢＹＴＥ的結果
LDA  $396   ⎪
JSR  $FDDA  ⎭
RTS           結束
```

多個BYTE的加法運算

我們都知道剛才的兩個ＢＹＴＥ加法可以處理四個位，而輸出則達到五個位，那麼我們如果需要多位有效數字時可以用多個ＢＹＴＥ來運算，以下是程式的例子，注意Ｘ暫存器內貯存所需運算的ＢＹＴＥ數，以下是四個ＢＹＴＥ的加法。

```
LDX   #$04
CLC
——ADD——
LDA   $320，X
ADC   $335，X
STA   $340，X
DEX
ONE  ADD
RTS
```

減法運算

減法運算和加法運算十分相似，我們的處理方法仍然是由右至左，6502 CPU會將借位考慮在內，我們用來作減法

加減乘除四則運算

運算的命令是SBC（SUBTRACT WITH CARRY），與ADC一樣，SBC可以用於立即模式，絕對模式，零頁位址模式及絕對索引位址模式之中，而且更可以用在一些其他的模式之中。

在處理減法運算之前，進位位元必須首先設定爲1，而這一點可以用SEC命令來達成，SEC（SET CARRY BIT）是一個隱含位址模式的命令，它只佔用一個BYTE的長度，以下的例字是將兩個分別爲兩個BYTE長的數字相加：

```
JSR  $FC58     清除螢幕
LDA  #$00    ⎫
STA  $394    ⎬ 將位址$394及$395 設定爲零以貯存
STA  $395    ⎭ 運算結果
SEC            設定進位位元
LDA  $391    ⎫ 計算LSB，它們分別在$391及$393
SBC  $393    ⎭ 位址之中
STA  $395      將剛才的結果貯存到$395之上
LDA  $390    ⎫ 計算MSB，它們分別在$390及$392
SBC  $392    ⎭ 位址之上
STA  $394      將MSB的運算結果貯存在$394之中
LDA  $394    ⎫ 顯示MSB的運算結果
JSR  $FDDA   ⎭
LDA  $395    ⎫ 顯示LSB的運算結果
JSR  $FDDA   ⎭
RTS            結束
```

以上的例子可以用以下的圖來說明：

```
$394 是MSB的運算結果          $395 是LSB的運算結果
   ↑                             ↑
$390 貯存第二個數的MSB        $391 貯存第一個數的LSB
   ＋                            ＋
$392 貯存第一個數的MSB        $393 貯存第一個數的LSB
```

（其中我們再重溫一次：

MSB＝MOST SIGNIFICANT BYTE最高位有效位元

LSB＝LEAST SIGNIFICANT BYTE最低位有效位元

而一個數字例如$ABCD之中$AB是MSB而$CD則是LSB，請各位緊記！）

乘法的運算

爲了要直接而有效地處理乘法運算，我們需要將移位及旋轉技術加以引用。爲了容易明白起見我先用紙筆的方法來引導各位，各位都知道57×36＝342＋171＝2052，但它經過三個步驟來達成：

```
①     57
     ×36
    ----
     342      用右下角的數字乘以在上方的數

②     57      用右左下角的數字乘以上方的數，但要向左
     ×36      移一個位
    ----
     342
    171

③     57      將兩列的數字相加以得到最後答案。
     ×36
    ----
     342
    171
    ----
    2052
```

若果我們用二進位數字來計算亦一樣：

57若以2進位數字表示則爲00111001

36若以2進位數字表示則爲00100100

兩數相乘：

```
           00111001
          ×00100100
       ----------------
           00000000     (1)
          00000000      (2)
         00111001       (3)
        00000000        (4)
       00000000         (5)
      00111001          (6)
     00000000           (7)
    00000000            (8)
   ----------------
    000100000000100
```

將得數換爲十進位數字則是2052

在以上的運算過程之中，其中只有第三行及第六行的運算才是有用的，因爲其它行裏的數字都是零，因此利用這種技術，可以得出以下的演算方式：

1 將$390，$391，$392 設置爲零以存入結果
2 將Y暫存器設爲零
3 增加Y
4 檢查Y內的數字（Y＝1代表要檢查下面最低行的數字）
5 若最低一行爲零則執行第七步
6 將移位後的行相加（$392及$393 的加到$390及$391）
7 將數字左移一行
8 檢查Y是否等於8
9 若Y<>8 則跳到第三步執行
10 印出結果
11 結束

每次經過這個程式執行之後，數字會經檢查底行是否爲零並加起來，由於要移位的關係，我們要採用一新命令及位址模式，它們是ASL及ROL和ROR，故此要採用較長的篇幅，我希望下一次再和各位討論，再見！■

dTEST

測驗你對dBASE Ⅱ的認識

香港電腦研習中心供稿

DBASE I I 是美國ASHTON－TATE 公司在一九八一年初出品的一種處理資料的軟件。（DBASE I 是不會存在的。ASHTON－TATE公司爲令人覺得產品是改良過及測試妥當的，所以一開始就取名DBASE I I 。）推出以來，被公認爲微型電腦實用軟件中的翹楚。甚至最新推出的產品如KNOWLEDGE MAN，R：BASE 等均以D－BASE I I 爲頭號假想敵，着着要與DBASE I I 比長短。由此可見DBASE I I 地位之尊。

學習DBASE I I 可分爲兩個階段。在第一個階段，學習DBASE I I 的指令及句法。你對DBASE I I 下一度指令，它便爲你做一件工作。你可以下一度指令，叫DBASE I I 計出全公司職員薪金的總數，只要字眼及句法正確，DBASE I I 便會照辦。在此階段你與DBASE I I 的溝通是INTERACTIVE 的——你踢一踢，它便郁一郁。

第二個階段是用DBASE I I 編寫PROGRAM。DB－ASE I I 本身是一個PROGRAM，但它亦是一種LA－NGUAGE，你可以用DBASE I I 來編寫適合你自己需要的PROGRAM。其實PROGRAM也者，就是一連串的指令。所謂寫PROGRAM，就是將一連串的指令儲起來，到需要時叫電腦照着它逐步去做。編寫PROGRAM比起INTERACTIVE地用COMMAND 有兩個好處：

一. 一件工作可能需要一連串的COMMAND去完成，若你把這些COMMAND連成一個PROCEDURE（也就是PROGRAM），你可以毋須坐在電腦旁邊等候它完成一度COMMAND，再下另一度COMMAND。你可以用一個迴圈（LOOP），叫它依次處理D－ATAFILE 中的每一個RECORD。在它工作的時候，你可以去喝茶、閒談，回來的時候工作已做妥當了。

二. 你可以根據自己的需要，編寫一套MENU－DRIVEN的資料處理系統，令到職員毋須學過DBASE I I ，也可以利用DBASE I I 處理資料。操作者毋須用CO－MMAND，一切工作都有MENU指引。

如果你對DBASE I I COMMAND 的運用，已到了應付裕如的程度，你便可以進一步學習用DBASE I I 編寫PROGRAM。

以下我們爲大家擬定了一個DBASE I I 的測驗，學過DBASE I I 的人可藉此了解自己對DBASE I I 的認識程度。沒有學過DBASE I I 的，亦可藉參看這個測驗，知道DBASE I I 用途的大概。

你將你公司職員的資料，記錄在一個DBASE I I 的檔案裏，檔案的結構如下：

每一個RECORD分爲8個FIELD，分别記職員的姓、名、任職部門、職級、職位、薪金、加入公司日期、年齡。

```
STRUCTURE FOR FILE:STAFF.DBF
NUMBER OF RECORDS: 00300
DATE OF LAST UPDATE:11/05/84
PRIMARY USE DATABASE
FLD    NAME      TYPE  WIDTH   DEC
001  SURNAME      C     008
002  FNAME        C     012
003  DEPT         C     010
004  GRADE        C     002
005  POSITION     C     016
006  SALARY       N     008     002
007  ENTRY        C     008
008  AGE          N     002
     TOTAL             00067
```

假定DATAFILE（STAFF.DBF）已被取到ME－MORY中（IN USE），你應用什麼指令去達到以下的效果：

1. 列出所有四級或以上任職於營業部的職員的名字。
2. 計算全公司職員一個月薪金的總數。
3. 董事會決定全公司職員加薪百分之十，將檔案內全部職員薪金資料按比例增加。
4. 董事會決定加薪幅度分爲兩種：四級或以上加薪百分之十，三級或以下加薪百分之十五。請在DATABASE內反映這項決定。
5. 爲這個DATAFILE 編一個INDEX FILE，先根據任職部門排列；若任職部門相同，根據姓排列；若姓相同，根據名排列。
6. 除了人事部職員外，所有職員到了六十五歲都要退休，請把到了退休年齡的職員的RECORD除去。
7. 數一數SALES DEPARTMENT 有多少職員。
8. INVENTORY DEPARTMENT 的總管向你索取該部門職員的資料，把該部門職員的資料抄一份給他。（抄到DRIVE C ：的磁碟中）
9. 計算職員的平均年齡。
10. 指出最高薪職員及最低薪職員薪金的差額。

答案及解說

1. LIST ALL SURNAME，FNAME FOR GRADE >=“4”. AND. DEPT=“SALES”
   由於4及SALES是文字——因GRADE及DEPT

dTEST——測驗你對 dBASE Ⅱ.的認識

都是CHARACTER FIELD，所以字串的前後均要加上引號。四級或以上，所以要用“>=”的符號，用“>”便錯了。

2. SUM SALARY
將DATABASE 內每一RECORD中SALARY項內的數目加起來。

3. REPLACE ALL SALARY WITH SALARY ＊1.1
將每個RECORD內的SALARY項的數值乘以1.1，再放回SALARY項內。REPLACE指令的結構是 REPLACE〔<SCOPE>〕<FIELDNAME> WITH<EXPRESSION>。
將WITH以後的資料取代WITH前指定的FIELD的內容。

4. REPLACE ALL SALARY WITH SALARY ＊1.1 FOR GRADE >= “4”
REPLACE ALL SALARY WITH SALARY ＊1.15 FOR GRADE <= “3”

5. INDEX ON DEPT＋SURNAME＋FNAME TO STAFFNM
字串＋字串，令到兩字串相連起來，‘A’＋‘B’的結果是‘AB’。INDEX的根據是DEPT＋SURNAME＋FNAME，即以三者連成一字串，再根據ALPHABETICAL ORDER 比大小。STAFFNM是INDEXFILE 的名稱。

6. DELETE ALL FOR AGE>=65.AND. DEPT <> ‘PERSONEL’
AGE是NUMERIC FIELD，所以65毋須用引號括起來。DELETE只把RECORD加上記號，若要從DATAFILE 除去，要再用PACK指令。

7. COUNT FOR DEPT= ‘SALES’

8. COPY TO C:INVSTAFF FOR DEPT= ‘INVENTORY’
如果你只把其中幾個FIELD抄給他，可以這樣寫：
COPY FIELD SURNAME，FNAME，DEPT，POSITION，AGE TO C:INVSTAFF FOR DEPT=‘INVENTORY’

9. SUM AGE TO TOTAL
GOTO BOTTOM
STORE # TO NO
STORE TOTAL／NO TO AV
? ‘AVERAGE AGE=’＋AV
將每一RECORD中AGE 欄內的VALUE加起來，總數存到TOTAL這個變數中。然後將POINTER移到檔案的最後一個RECORD，將其RECORD NUMBER（#）存到變數NO之中，將總數（TOTAL）除以人數（NO）得到平均年齡（AV），然後顯示出來。

10. INDEX ON SALARY TO STAFFSAL
USE STAFF INDEX STAFFSAL
GOTO TOP
STORE SALARY TO MIN
GOTO BOTTOM
STORE SALARY TO MAX
? MAX－MIN
根據SALARY的多少編一個INDEX FILE，USE STAFF INDEX STAFFSAL將DATAFILE內的RECORD根據STAFFSAL這個INDEX FILE排列，也就是說，根據SALARY排列。由於排列是IN ASCENDING ORDER，第一個RECORD的 SALARY 是最低薪職員的人工，將這個數目存到變數MIN裏；最後的一個RECORD的SALARY是最高薪職員的人工，將這個數目存到變數MAX之中。MAX減MIN便是差額。■

TAB & X Y

許多時，在編寫程式時，我們會經常用到HTAB X ：VTAB Y 來設定螢幕輸出。你是否對打入這兩個指令感到厭倦？這裏爲你提供一個簡短的程式幫助你代替打入上述的指令。

假若你想螢幕輸出格式爲HTAB 10：VTAB 15 ，你只須簡單的使用& 10，15來代替打入HTAB 10：VTAB 15便可。

這個程式使用的語法是& X ，Y 。X 可由1至40（如果是80字行顯示，則X 可由1至80），Y 值則一定是由1至24。假如輸入的數值超出範圍，則電腦會有“嘟”聲響告知你輸入錯誤。

```
03A4- A9 4C 8D F5
03A8- 03 A9 B4 8D F6 03 A9 03
03B0- 8D F7 03 60 20 F8 E6 CA
03B8- 86 24 20 4C E7 CA E0 18
03C0- B0 0A 8A A6 24 EC 50 B0
03C8- 03 4C 5B FB 4C 3A FF
*
```

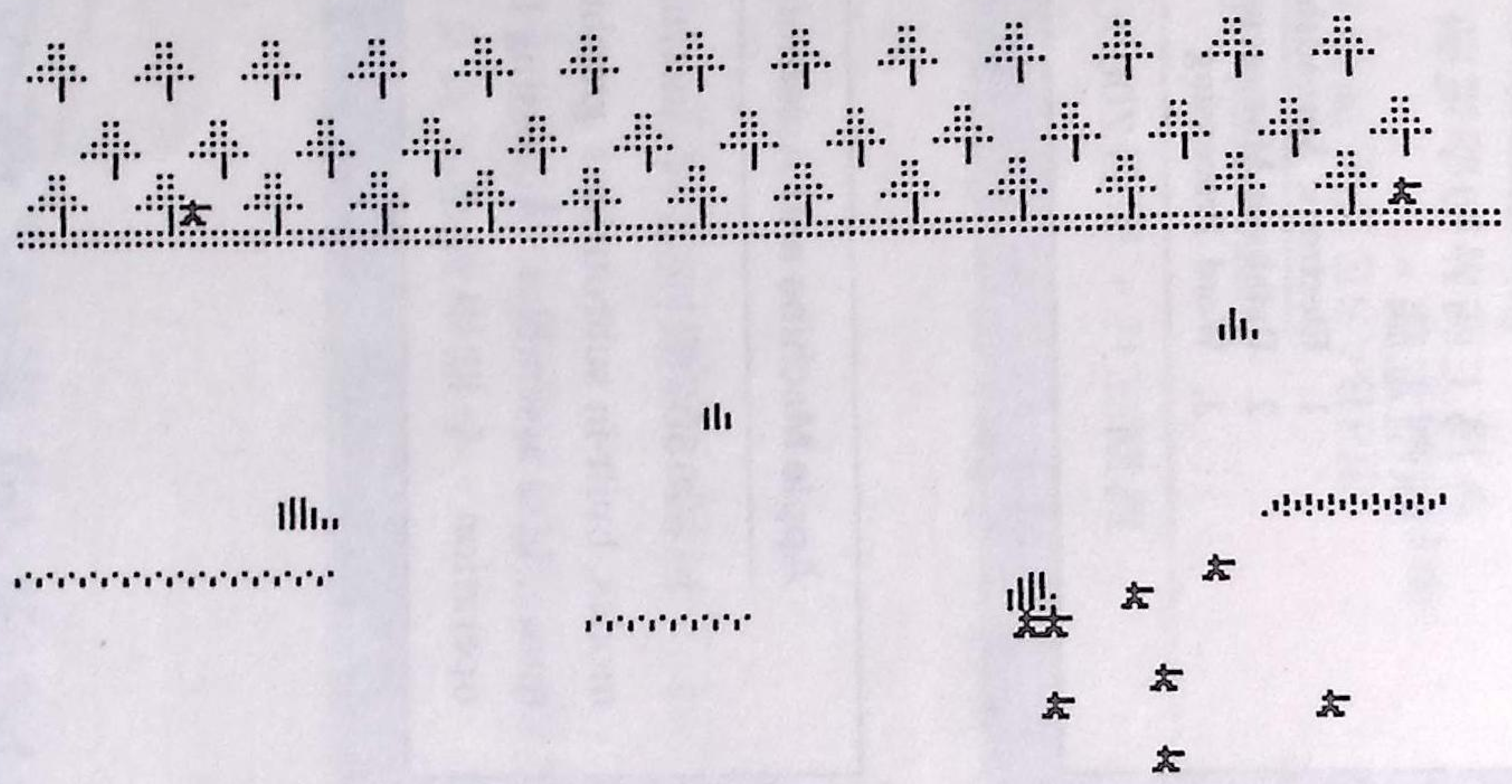
ILL. ERROR
99 F1

使用這個& X ，Y 程式時，留意必須要同時輸入X ，Y 值，同時要先RUN & X ，Y ，方可在編寫的程式中使用。

本程式在程式磁碟的FILENAME是TAB & X，Y 。■

模擬動態戰爭遊戲

諾曼第登陸戰



上期爲大家介紹的遊戲JACKIES HEAD相信大家都會覺得有趣。今期要爲大家介紹的是一個截然不同的遊戲——諾曼第登陸戰。這是一個模擬戰爭的遊戲，既要考驗閣下的戰略運用，又要碰運氣。身爲指揮官的閣下擁有一小縱隊士兵，爲盟軍執行一項重要的任務，率先登岸想辦法摧毀敵人隱藏在山崖上的一門大砲，以便讓隨後的盟軍援兵安全登陸。

由於要避免被敵人發現，你的登陸步隊只能携帶少量的軍備武器，包括一支來福槍及兩枚手榴彈。好不幸，你的敵人擁有的武器無論火力及數量均遠較你爲多，其中包括：

(1) 一門大砲
(2) 狙擊兵
(3) 迫擊砲

而且敵人更在海灘四處佈有鐵線網及地雷陣，每一項武器都足以摧毀你的下屬。

遊戲開始時，你的小縱隊人馬（十二人）會登上灘頭，由電腦控制的敵方會即時發射開火，其次序是迫擊砲、跟著是大砲，然後由狙擊兵發射槍砲。

之後你可指揮各人快速前進（用R鍵）或伏地爬行（用C鍵）。I、J、K及M鍵是用來控制跑或爬行的方向。當所有士兵行動部署好後，又會輪到電腦來控制敵人發砲。

當然，控制遊戲進行還有許多按鍵，包括：

D——伏下
U——站立
A——瞄準來福槍
F——發射來福槍
L——裝彈藥
P——拉手榴彈保險掣
T——擲手榴彈
?——參考各指令按鍵
Q——離開遊戲

假如你不知道那個是要接受指揮命令的士兵，除了上述鍵外，隨意按任何鍵都會令該士兵所處位置閃耀。

S 鍵是决定該士兵在每一個回合中

```
10  REM  ********************
20  REM  **                **
30  REM  **  NORMANDY WAR  **
40  REM  **       BY       **
50  REM  **   COMPUHOUSE   **
60  REM  **                **
70  REM  ********************
80  IF  PEEK (104) <  > 64 THEN  POKE
     103,1: POKE 104,64: POKE 163
     84,0: PRINT  CHR$ (4);"RUN N
     ORMANDY WAR"
90  REM  ACTUAL START OF PROGRAM
     AFTER LOADING ABOVE HI-RES S
     CREEN
100  POKE 232,1: POKE 233,8
110  PRINT  CHR$ (4);"BLOAD NORMA
     NDY BEACH,A$801"
120  HOME : VTAB 10: HTAB 15: PRINT
     "NORMANDY WAR": PRINT
150  DIM A(12,2),M(20,2),PO$(12),
     ST$(12),SH(12),G(12),FF(12),
     EE(12)
160  VTAB 20: HTAB 10: INPUT "INS
     TRUCTIONS? (Y/N) ";X$
170  IF X$ = "Y" THEN 3080
180  IF X$ <  > "N" THEN 160
190  :
200  REM   **  SET VARIABLES  **
210  :
220  FOR X = 1 TO 5: REM  MINES
230 M(X,1) = ( INT ( RND (1) * 9)
     + 12) * 5:M(X,2) = 110
240  NEXT X
250  FOR X = 6 TO 15
260 M(X,1) = ( INT ( RND (1) * 20
     ) + 27) * 5:M(X,2) = 110
270  NEXT X
280  FOR X = 16 TO 20
290 M(X,1) = ( INT ( RND (1) * 8)
     + 47) * 5:M(X,2) = 110
300  NEXT X
310  FOR X = 1 TO 12
320 ST$(X) = "OK":PO$(X) = "R":SH
     (X) = 1:G(X) = 2:FF(X) = 0:E
     E(X) = 0
330  NEXT X
340 MG(1) =  INT ( RND (1) * 277)
     + 1: REM  MACHINE GUN
350 MG(2) = 40
360 FL = 0:RG = 30:LL = 1:RR = 1:
     GG = 1
370  HOME : VTAB 10
380  PRINT "LEVEL OF DIFFICULTY:"
390  PRINT : PRINT "1) VERY DIFFI
     CULT"
400  PRINT "2) ALMOST IMPOSSIBLE"
410  PRINT : INPUT "PLEASE ENTER
     YOUR CHOICE (1 OR 2) ";ZZ$
420  IF ZZ$ <  > "1" AND ZZ$ <  >
     "2" THEN 370
430 VV =  VAL (ZZ$)
440  GOTO 960
450  :
460  REM   **  SUB-ROUTINES  **
470  :
480 ZZ =  PEEK ( - 16336):ZZ =  PEEK
     ( - 16336):ZZ =  PEEK ( - 16
     336): RETURN : REM  SOUND
490  IF FL = 1 THEN  XDRAW 2 AT X
     C,YC: REM  EXPLOSIONS
500  IF FL = 2 THEN  XDRAW 2 AT A
     (X,1),A(X,2)
510  FOR Z = 1 TO 500: NEXT Z
520  IF FL = 1 THEN  XDRAW 2 AT X
     C,YC
530  IF FL = 2 THEN  XDRAW 2 AT A
     (X,1),A(X,2)
540  HCOLOR= 0: DRAW 1 AT A(X,1),
     A(X,2)
550  HCOLOR= 3:A(X,1) = 0:A(X,2) =
     0
560 ST$(X) = "KILLED"
570  HOME : VTAB 23
580  IF FL = 1 THEN  PRINT "MAN #
     ";X;" KILLED BY MINE."
590  IF FL = 2 THEN  PRINT "MAN #
     ";X;" BLOWS HIMSELF UP!"
600  FOR Z = 1 TO 1000: NEXT Z
610  HOME : VTAB 23: RETURN
620  FOR E = 1 TO 200: NEXT E: RETURN
630  IF C$ = "D" THEN FF(X) = 1
640  IF C$ = "U" THEN FF(X) = 0
650  IF C$ = "L" THEN SH(X) = 1: GOTO
     760
660  IF C$ = "A" AND PO$(X) = "R"
     THEN  HOME : VTAB 23: PRINT
     "YOU CAN'T AIM WHILE RUNNING
     .": FOR E = 1 TO 1000: NEXT
     E: HOME : VTAB 23:FL = 3: RETURN
```

最後所處於的位置。

士兵的來福槍是用來對付狙擊手，而手榴彈是用來摧毀大砲的。初時，大砲所處的位置是隱蔽的，不過它是安置在山崖上。你及你的隊伍必須衝破地雷陣及砲火網來深入敵兵陣地。當然，越接近敵人後防，受擊的危險性越大。不過灘頭上有零散的石頭作爲掩護。而且假如士兵是伏地爬行（伏地爬行每次只能移動一步），則可避過大砲的射擊。當然，留在沙灘時間越久，受迫擊砲擊中的機會越大。

遊戲進行時，螢幕會分爲兩部份，上半部是戰場實況，下半部是指出各士兵所處於的狀態，其中包括：

(1)每個士兵所處的狀態（即你給予指示的最後一個指示，例如R表示在跑）
(2)每個士兵的位置（站立或伏地）
(3)來福槍剩餘的子彈發數
(4)手榴彈剩餘數目

當敵方還擊時，電腦會告訴誰人中彈身亡。

留意遊戲進行時，每一回合，當所有士兵接受完行動訓令時，電腦控制的敵方會立即發砲。然後顯示死亡人數。然後再重新一個回合的指示安排。

在擲手榴彈前，要先按P拉下保險掣，才可擲出。同時，在按P之前電腦會告訴你大砲是否在視野範圍之內，大砲必須要在視野範圍內，否則是不能拉動保險掣，縱然是擲出手榴彈，但都屬於白花彈藥。

步兵在走動時，可隨意中途改變方向繼續移動，如此做最多可行6步。但在爬行（C）狀態下，無論那個方向都只能走一步。

灘頭上的地雷陣是隱蔽的，並不容易發現，玩者宜小心。

要射擊敵方狙擊手，無須深入敵方陣地，只在岸邊亦可射擊，但必要先伏下（D）來，才能瞄準（A），然後發射。發射前，電腦會詢問目的物的方位。

由於敵方是受電腦控制，所以任務會變得非常困難，遊戲開始時有兩個程度供選擇：(1)非常困難，(2)絕沒有可能成功。

爲了讓大家學習這遊戲的編寫投巧，筆者現將程式每部份分析。

行句80將主程式移至高解像頁地址

```
670  IF C$ = "F" AND SH(X) > 0 AND
     PO$(X) <  > "A" THEN  HOME :
      VTAB 23: PRINT "YOU JUST SH
     OT YOURSELF IN THE FOOT.": PRINT
     "YOU DIDN'T AIM FIRST.": FOR
     E = 1 TO 1500: NEXT E:SH(X) =
     SH(X) - 1:PO$(X) = C$: RETURN
680  IF C$ = "F" AND SH(X) = 0 THEN
      HOME : VTAB 23: PRINT "CLIC
     K - RIFLE NOT LOADED!": FOR
     E = 1 TO 1000: NEXT E:PO$(X)
      = C$: RETURN
690  IF C$ = "F" THEN PO$(X) = C$
     : GOTO 2730
700  IF PO$(X) = "P" AND C$ <  >
     "T" THEN FL = 2: GOSUB 450: GOSUB
     490:PO$(X) = C$: RETURN
710  IF C$ = "P" AND G(X) = 0 THEN
      HOME : VTAB 23: PRINT "NO G
     RENADES LEFT.": FOR E = 1 TO
     1000: NEXT E: HOME : VTAB 23
     :FL = 3: RETURN
720  IF C$ = "T" AND G(X) = 0 THEN
      HOME : VTAB 23: PRINT "NO G
     RENADES LEFT.": FOR E = 1 TO
     1000: NEXT E: HOME : VTAB 23
     :FL = 3: RETURN
730  IF C$ = "T" AND PO$(X) <  >
     "P" THEN  HOME : VTAB 23: PRINT
     "NICE THROW": FOR E = 1 TO 1
     000: NEXT E: PRINT " - BUT Y
     OU FORGOT TO PULL THE PIN": FOR
     E = 1 TO 1000: NEXT E:G(X) =
     G(X) - 1:EE(X) = 1: HOME : VTAB
     23:PO$(X) = C$: RETURN
740  IF C$ = "P" AND A(X,2) =  >
     85 AND A(X,1) > 1 THEN  HOME
     : VTAB 23: PRINT "CAN'T SEE
     THE MACHINE GUN FROM HERE,SI
     R.": FOR E = 1 TO 1500: NEXT
     E:FL = 3: HOME : VTAB 23: RETURN
750  IF C$ = "T" THEN PO$(X) = C$
     : GOTO 2960
760 PO$(X) = C$: RETURN
770  HCOLOR= 0: DRAW 1 AT XO,YO: HCOLOR=
     3: DRAW 1 AT XC,YC:PO$(X) = "STU
     CK":A(X,1) = XC:A(X,2) = YC:
      RETURN
780  IF A(X,2) =  > 60 AND A(X,2)
      =  < 70 AND A(X,1) =  > 225
      AND A(X,1) =  < 235 THEN AA
      = 1: RETURN
790  IF A(X,2) =  > 115 AND A(X,2
     ) =  < 125 AND A(X,1) =  > 1
     90 AND A(X,1) =  < 200 THEN
     AA = 1: RETURN
800  IF A(X,2) =  > 80 AND A(X,2)
      =  < 90 AND A(X,1) =  > 130
      AND A(X,1) =  < 140 THEN AA
      = 1: RETURN
810  IF A(X,2) =  > 95 AND A(X,2)
      =  < 105 AND A(X,1) =  > 50
      AND A(X,1) =  < 65 THEN AA =
     1: RETURN
820  RETURN
830  HOME : NORMAL : VTAB 23: PRINT
     "NICE THROW": FOR E = 1 TO 1
     000: NEXT E: XDRAW 2 AT D1,D
     2
840  GOSUB 450
850  HOME : NORMAL : VTAB 24: PRINT
     "NOT WITHIN OUR RANGE, SIR."

860  FOR E = 1 TO 1000: NEXT E
870  XDRAW 2 AT D1,D2
880  RETURN
890  HPLOT X - 1,Y TO X + 1,Y: REM
      TREE SHAPE
900  HPLOT X - 2,Y + 2 TO X + 2,Y
      + 2
910  HPLOT X - 4,Y + 4 TO X + 4,Y
      + 4
920  HPLOT X - 6,Y + 6 TO X + 6,Y
      + 6
930  HPLOT X + 1,Y + 6 TO X + 1,Y
      + 10
940  RETURN
950  HPLOT X,Y TO X + 4,Y - 3: RETURN

960 :
970  REM   ** DISPLAY MAP OF ARE
     A  **
980 :
990  HGR
1000  HCOLOR= 1: SCALE= 1: ROT= 0

1010  FOR X = 10 TO 250 STEP 20: REM
      TREES
1020 Y = 5: GOSUB 890: NEXT X
1030  FOR X = 20 TO 260 STEP 20
1040 Y = 20: GOSUB 890: NEXT X
1050  FOR X = 10 TO 250 STEP 20
1060 Y = 30: GOSUB 890: NEXT X
1070  HCOLOR= 5: REM  ROCKS
1080  HPLOT 228,59 TO 229,59 TO 2
     30,60 TO 228,60 TO 228,61 TO
     231,61 TO 231,62 TO 227,62 TO
     227,63 TO 232,63 TO 233,64 TO
     226,64
1090  HPLOT 190,108 TO 192,108 TO
     193,109 TO 189,110 TO 193,11
     0 TO 194,111 TO 188,111 TO 1
     87,112 TO 195,112 TO 186,114
      TO 196,114
1100  HPLOT 132,75 TO 133,75 TO 1
     34,76 TO 132,76 TO 131,77 TO
     134,77 TO 135,78 TO 131,78 TO
     130,79 TO 136,79
1110  HPLOT 52,92 TO 54,92 TO 55,
     93 TO 51,94 TO 57,94 TO 58,9
     5 TO 50,95 TO 50,96 TO 60,96
      TO 62,97 TO 50,97
1120  HCOLOR= 2
1130  HCOLOR= 2: FOR X = 107 TO 1
     35 STEP 4
1140 Y = 117: GOSUB 950: NEXT X: REM
      BARBED WIRE
1150  FOR X = 234 TO 265 STEP 4
1160 Y = 97: GOSUB 950: NEXT X
1170  FOR X = 1 TO 60 STEP 4
1180 Y = 108: GOSUB 950: NEXT X
1190  HCOLOR= 3: HPLOT 2,150 TO 2
     78,150: REM  WAVES
1200  HCOLOR= 6: HPLOT 2,153 TO 2
     78,153
1210  HPLOT 2,156 TO 278,156
1220  HCOLOR= 5: HPLOT 2,41 TO 27
     8,41: HPLOT 2,43 TO 278,43: REM
      CLIFFS
1230 :
1240  REM    **  SHOW MEN  **
1250 :
1260  HCOLOR= 3:I = ( INT ( RND (
     1) * 46) + 5) * 5
1270  DRAW 1 AT I - 10,125:A(1,1)
      = I - 10:A(1,2) = 125: GOSUB
     620
1280  DRAW 1 AT I,125:A(2,1) = I:
     A(2,2) = 125: GOSUB 620
1290  DRAW 1 AT I + 10,125:A(3,1)
      = I + 10:A(3,2) = 125: GOSUB
     620
1300  DRAW 1 AT I - 10,130:A(4,1)
      = I - 10:A(4,2) = 130: GOSUB
     620
1310  DRAW 1 AT I,130:A(5,1) = I:
     A(5,2) = 130: GOSUB 620
1320  DRAW 1 AT I + 10,130:A(6,1)
      = I + 10:A(6,2) = 130: GOSUB
     620
1330  DRAW 1 AT I - 10,135:A(7,1)
      = I - 10:A(7,2) = 135: GOSUB
     620
1340  DRAW 1 AT I,135:A(8,1) = I:
     A(8,2) = 135: GOSUB 620
1350  DRAW 1 AT I + 10,135:A(9,1)
      = I + 10:A(9,2) = 135: GOSUB
     620
1360  DRAW 1 AT I - 10,140:A(10,1
     ) = I - 10:A(10,2) = 140: GOSUB
     620
1370  DRAW 1 AT I,140:A(11,1) = I
     :A(11,2) = 140: GOSUB 620
```

諸曼第登陸戰

之上，即$4001位址，此舉是避免程式在高解像第一頁執行。

行句100至110將設定指標及載入所要的圖像。其中包括人像，及爆炸圖案。

行句150將程式中要用到的幾個變數設定DIMENSION。人及地雷是採用二次元陣列。例如A（12，2）可以有最多12個人及每個人有X，Y座標。

行句200－360 設定變數的值。行句220－300 隨機放置地雷；行句340至350 隨機放置大砲。表(1)列出所有變數及旗號。

行句460－940是一個令遊戲較快速進行的子程式，當然，這類遊戲速度並不重要。

行句480產生每一次發砲的聲响。

行句490－610 XDRAW 手榴彈及地雷的爆炸。利用這子程式，爆炸的圖像就不會破壞背境圖案。同時，這段程式亦利用到HCOLOR＝0及DRAW將被擊死的人消除，及在螢幕底部打出顯示死者是誰人。

行句630－820爲所有的指令（除了R及C外）提供IF－THEN指述。

行句830－850是用來測試擲出手榴彈後的反應，同時將結果顯示。

行句890－940 繪畫出在山崖上的矮樹。

行句970－1220製造出戰場。

行句1240－1420將士兵登陸位置隨機決定。

行句1440－2030是主迴路所在。所有指揮訓令及人物的移動都是在此決定。行句1490至1590顯示各人的所處狀態及接受訓令。行句1600至1640利用XD－RAW 閃耀士兵所站位置。

行句1750至2000是移動方向（I,J,M及J鍵 ）的控制。開始位址是Xo及Yo ，而即時移動位址是XC及YC。

行句1650（及1810）檢驗有否按鍵輸入，假如有，存於－16384的數值便會大於127 。欲要檢查第二個按鍵輸入，必須用POKE－16368，0 敍述句將－16384 位址值重置。

行句1850－1870檢查士兵有否行入佈雷陣，如有，就跳至爆炸的子程序。

行句1970量度士兵移動的步數及規限可容許的最大移動距離。

行句2050－2280是迫擊砲部份。行

```
1380  DRAW 1 AT I + 10,140:A(12,1
 ) = I + 10:A(12,2) = 140: GOSUB
 620
1390  HOME : VTAB 23: PRINT "LAND
 ING COMPLETED.": FOR Z = 1 TO
 1000: NEXT Z
1400  DRAW 1 AT 35,35
1410  DRAW 1 AT 260,35
1420  GOTO 2040
1430 :
1440. REM   ** MOVEMENT PHASE **
1450 :
1460  FOR X = 1 TO 12
1470 FL = 0
1480  IF ST$(X) = "KILLED" THEN 2
 030
1490  HOME : VTAB 22: PRINT "MAN
 #";
1500  PRINT  TAB( 7)"STATUS";
1510  PRINT  TAB( 14)"POSITION";
1520  PRINT  TAB( 23)"SHOTS";
1530  PRINT  TAB( 30)"GRENADES"
1540  PRINT  TAB( 3)X;
1550  PRINT  TAB( 8)PO$(X);
1560  IF FF(X) = 0 THEN  PRINT  TAB(
 16)"UP";
1570  IF FF(X) =1 THEN  PRINT  TAB(
 16)"DOWN";
1580  PRINT  TAB( 25)SH(X);
1590  PRINT  TAB( 32)G(X)
1600  FOR Y = 1 TO 10
1610  XDRAW 1 AT A(X,1),A(X,2)
1620  FOR E = 1 TO 10: NEXT E
1630  XDRAW 1 AT A(X,1),A(X,2)
1640  NEXT Y
1650 I =  PEEK ( - 16384): IF I >
 127 THEN  POKE  - 16368,0
1660  PRINT "COMMAND? (ENTER '?'
 TO SEE LEGEND )";: GET C$
1670  IF C$ <  > "R" AND C$ <  >
 "C" AND C$ <  > "D" AND C$ <
  > "U" AND C$ <  > "A" AND C
 $ <  > "F" AND C$ <  > "L" AND
 C$ <  > "P" AND C$ <  > "T" AND
 C$ <  > "?" AND C$ <  > "Q" THEN
 1490
1680  IF C$ = "Q" THEN 3820
1690  IF C$ = "?" THEN  HOME : VTAB
 21: PRINT "C-CRAWL  R-RUN  S
 -SET POSITION  A-AIM": PRINT
 "F-FIRE  L-LOAD   D-LIE DOWN
   U-STANDUP": PRINT "P-PULL
 PIN   T-THROW GRENADE  Q-QU
 IT": INPUT "PRESS RETURN TO
 CONTINUE..";Z$: IF Z$ = "" THEN
 HOME : VTAB 23: GOTO 1490
1700  IF PO$(X) = "P" AND C$ <  >
 "T" THEN FL = 2: GOSUB 450: GOSUB
 490: GOTO 2030
1710  IF C$ <  > "R" AND C$ <  >
 "C" THEN  GOSUB 630: IF FL =
 3 THEN FL = 0: GOTO 1490
1720  IF C$ <  > "R" AND C$ <  >
 "C" THEN 2030
1730  IF C$ = "R" AND FF(X) = 1 THEN
 HOME : VTAB 23: PRINT "YOU
 MUST STANDUP BEFORE RUNNING"
 : FOR E = 1 TO 1000: NEXT E:
 GOTO 1490
1740  IF C$ = "C" AND FF(X) = 0 THEN
 HOME : VTAB 23: PRINT "YOU
 MUST LIE DOWN BEFORE CRAWLIN
 G": FOR E = 1 TO 1000: NEXT
 E: GOTO 1490
1750 XC = A(X,1):YC = A(X,2)
1760 XO = XC:YO = YC
1770  IF C$ = "C" THEN ST = 5:PO$
 (X) = C$
1780  IF C$ = "R" THEN ST = 20:PO
 $(X) = C$
1790 K =  PEEK ( - 16384): IF K >
 127 THEN  POKE  - 16368,0
1800  XDRAW 1 AT XC,YC
1810  IF K < 128 THEN K$ = "": GOTO
 1830
1820 K$ =  CHR$ (K - 128)
1830  IF K$ = "I" THEN YC = YC -
 5
1840  IF YC <  > 110 THEN 1880
1850  FOR U = 1 TO 20
1860  IF XC = M(U,1) THEN FL = 1:
 GOSUB 450: GOSUB 490:M(U,1)
 = 0:M(U,2) = 0: GOTO 2030: REM
 MINE EXPLOSION
1870  NEXT U
1880  IF YC = 115 AND XC > 110 AND
 XC < 135 THEN YC = 115: GOSUB
 770: GOTO 2030: REM  BARBED
 WIRE
1890  IF YC = 95 AND XC > 235 AND
 XC < 265 THEN YC = 95: GOSUB
 770: GOTO 2030
1900  IF YC = 105 AND XC > 2 AND
 XC < 60 THEN YC = 105: GOSUB
 770: GOTO 2030
1910  IF YC =  < 40 THEN  HOME : VTAB
 23: PRINT "FELL OFF THE CLIF
 F - TOUGH LUCK.": FOR E = 1 TO
 2000: NEXT E: HCOLOR= 0: DRAW
 1 AT XO,YO: DRAW 1 AT XC,YC:
 HCOLOR= 3:ST$(X) = "KILLED"
 :A(X,1) = 0:A(X,2) = 0: GOTO
 2030
1920  IF K$ = "J" THEN XC = XC -
 5
1930  IF XC =  < 5 THEN FL = 1: GOSUB
 450: GOSUB 490: GOTO 2030
1940  IF K$ = "K" THEN XC = XC +
 5
1950  IF XC =  > 275 THEN FL = 1:
 GOSUB 450: GOSUB 490: GOTO
 2030
1960  IF K$ = "M" THEN YC = YC +
 5
1970  IF  INT ( SQR ((YC - YO) ^
 2 + (XC - XO) ^ 2)) >  = ST THEN
  HCOLOR= 0: DRAW 1 AT XO,YO:
  HCOLOR= 3: DRAW 1 AT XC,YC:
 A(X,1) = XC:A(X,2) = YC: HOME
 : VTAB 23: PRINT "YOU ARE ON
 LY ALLOWED ";ST / 5;" PACES"
 : FOR E = 1 TO 1000: NEXT E:
  HOME : GOTO 2030
1980  IF K$ = "S" THEN  HCOLOR= 0
 : DRAW 1 AT XO,YO: HCOLOR= 3
 : DRAW 1 AT XC,YC:PO$(X) = C
 $:A(X,1) = XC:A(X,2) = YC: GOTO
 2030
1990  IF YC =  > 150 THEN  HOME :
  VTAB 23: PRINT "MAN #";X;"
 DROWNED IN THE SEA": FOR E =
 1 TO 1000: NEXT E: HCOLOR= 0
 : DRAW 1 AT XO,YO: DRAW 1 AT
 XC,YC: HCOLOR= 3:ST$(X) = "K
 ILLED":A(X,1) = 0:A(X,2) = 0
 : GOTO 2030
2000  XDRAW 1 AT XC,YC
2010  GOTO 1790
2020 PO$(X) = C$
2030  NEXT X
2040 :
2050  REM   **  MORTAR SHELL PHAS
 E  **
2060 :
2070  HOME : VTAB 23: INVERSE :TT
  = 0:TX = 0:TY = 0
2080  PRINT "ENEMY MORTAR FIRE"
2090  FOR X = 1 TO 12
2100  IF ST$(X) = "KILLED" THEN 2
 130
2110 TX = TX + A(X,1):TY = TY + A
 (X,2): REM  AVG. OF POSITION
 S
2120 TT = TT + 1
2130  NEXT X
2140  IF TT =  < 0 THEN 3820
2150 HH =  INT (TX / TT):SS =  INT
 (TY / TT)
2160  FOR E = 1 TO 2
2170 MX = HH +  INT ( RND (1) * 5
 0) -  INT ( RND (1) * 50):MY
 = SS +  INT ( RND (1) * 50)
 -  INT ( RND (1) * 50)
2180  IF MX > 275 OR MX < 5 OR MY
 > 150 OR MY < 60 THEN 2170
```

句2110至2150將所有人的位置X，Y值加上來然後平均除（即HH及SS）。行號2150至2220在X，Y平均值的位址上爆炸。行句2230至2270利用PYTHA—GOREAN 定理計算每個士兵離開砲彈落點的距離（DMS）。如果距離少於6個位置，該士兵便算做被擊中。

行句2300—2550是敵人大砲的子程序部份。行句2400至2420模仿大砲的發射。行句2460令受到保護的士兵免於被轟。行句2470至2530決定轟中士兵的機會率（DR）—這同時決定於遊戲的難度、士兵是否站立或伏下、與大砲的距離。假如DR=＜3，該士兵便算被擊中。

行句2570—2700是敵兵開槍的子程序。同樣是利用DR來計算士兵被狙擊手槍火射殺的機會。

行句2740—2810是盟軍攻擊的部份。行句2770—2810詢問選擇目標（左或右）。電腦會以士兵距離來計算敵方狙

表(1) 變數及旗號

A（12，2）	士兵的編號及所在X　Y位址。
M（20，2）	每個地雷陣之號數及X、Y位址。
PO$（12）	每個士兵的最後所接受的訓令。
ST$（12）	每個士兵的狀況（生或死）。
SH（12）	槍膛上子彈的發數。
G（12）	每個持有手榴彈的數目。
FF（12）	每個的位置（站立或伏地）。
EE（12）	用來將未拔除保險制而擲出之手榴彈爆炸取消的旗號。
FL	用來指出士兵是否被地雷或大砲轟中的旗號。
RG	手榴彈爆炸的範圍。
LL	左邊狙擊手的生死情形。
RR	右邊狙擊手的生死情況。
GG	大砲的存在情況。
ST	士兵的移動步數。
AA	指示士兵是否在大石後受掩護的旗號。
DF	用來指出是否能看見大砲的旗號。

```
2190  XDRAW 2 AT MX,MY
2200  GOSUB 450
2210  FOR Z = 1 TO 500: NEXT Z
2220  XDRAW 2 AT MX,MY
2230  FOR X = 1 TO 12
2240  IF PO$(X) = "KILLED" THEN 2
   270
2250 DMS =  INT ( SQR ((A(X,1) -
   MX) ^ 2 + (A(X,2) - MY) ^ 2)
   ): REM  DISTANCE FROM EXPLOS
   ION
2260  IF DMS < 6 THEN  VTAB 24: NORMAL
   : PRINT "MAN #";X;" KILLED B
   Y MORTAR FIRE": FOR Z = 1 TO
   1000: NEXT Z: HCOLOR= 0: DRAW
   1 AT A(X,1),A(X,2): HCOLOR=
   3:ST$(X) = "KILLED":A(X,1) =
   0:A(X,2) = 0
2270  NEXT X
2280  NEXT E: NORMAL
2290 :
2300  REM  * ENEMY MACHINE GUN *
2310 :
2320 DF = 0: IF GG = 0 THEN 3670
2330  HOME : VTAB 23: INVERSE : PRINT
   "ENEMY MACHINE GUN FIRE"
2340  FOR X = 1 TO 12
2350  IF A(X,2) < 85 AND A(X,1) >
   1 THEN DF = 1: REM  CAN SEE
   MACHINE GUN
2360  GOSUB 450
2370  NEXT X
2380  IF DF = 0 THEN 2430
2390  NORMAL : VTAB 24: PRINT "WE
   CAN SEE MACHINE GUN, SIR!"
2400  FOR Y = 1 TO 30
2410  HCOLOR= 3: HPLOT MG(1),MG(2
   ) TO MG(1),MG(2) + 2: FOR E =
   1 TO 10: NEXT E: HCOLOR= 0: HPLOT
   MG(1),MG(2) TO MG(1),MG(2) +
   2: FOR E = 1 TO 10: NEXT E
2420  NEXT Y
2430  FOR X = 1 TO 12
2440  IF ST$(X) = "KILLED" THEN 2
   540
2450  IF A(X,2) =  < 50 THEN 2540
2460 AA = 0: GOSUB 780: IF AA = 1
   THEN 2540: REM  ROCK COVER
2470  IF VV = 1 THEN DR =  INT ( RND
   (1) * 15) + 1: REM  DIE ROLL
2480  IF VV = 2 THEN DR =  INT ( RND
   (1) * 9) + 1: REM  DIE ROLL
2490  IF FF(X) = 1 THEN DR = DR +
   2
2500  IF A(X,2) > 100 THEN DR = D
   R + 1
2510  IF A(X,2) =  < 100 THEN DR =
   DR - 2
2520  IF A(X,2) < 80 THEN DR = DR
   - 2
2530  IF DR =  < 3 THEN  NORMAL :
   VTAB 24: PRINT "MAN #";X;"
   KILLED BY MACHINE GUN FIRE":
   FOR Z = 1 TO 1000: NEXT Z: HCOLOR= 0:
      DRAW 1 AT A(X,1),A(X,2): HCOLOR=
   3:ST$(X) = "KILLED":A(X,1) =
   0:A(X,2) = 0
2540  NEXT X
2550  NORMAL
2560 :
2570  REM  * DEFENDER FIRING *
2580 :
2590  IF LL = 0 AND RR = 0 THEN 2
   710
2600  HOME : VTAB 22: INVERSE : PRINT
   "ENEMY SNIPER FIRE": FOR E =
   1 TO 500: NEXT E
2610  FOR X = 1 TO 12
2620  IF ST$(X) = "KILLED" THEN 2
   700
2630  IF A(X,2) =  > 100 THEN 270
   0
2640 AA = 0: GOSUB 780: IF AA = 1
   THEN 2700: REM  ROCK COVER
2650 DR =  INT ( RND (1) * 6) + 1
   : REM  DIE ROLL
2660  IF A(X,2) < 80 THEN DR = DR
   - 1
2670  IF A(X,2) < 60 THEN DR = 1
2680  GOSUB 450
2690  IF DR =  < 2 THEN  NORMAL :
   VTAB 24: PRINT "MAN #";X;"
   KILLED BY SNIPER FIRE": FOR
   Z = 1 TO 1500: NEXT Z: HCOLOR=
   0: DRAW 1 AT A(X,1),A(X,2): HCOLOR=
   3:ST$(X) = "KILLED":A(X,1) = 0:A
   (X,2) = 0
2700  NEXT X
2710  NORMAL
2720  GOTO 1430
2730 :
2740  REM  ** ATTACKER FIRING **
2750 :
2760  NORMAL : HOME : VTAB 23
2770  IF LL = 0 AND RR = 0 THEN  PRINT
   "BOTH SNIPERS ARE ALREADY KI
   LLED, SIR?": FOR E = 1 TO 10
   00: NEXT E: GOTO 760
2780  PRINT "ENTER TARGET FOR MAN
   #";X
2790  INPUT "L)EFT SNIPER OR R)IG
   HT SNIPER? ";ZZ$
2800  IF ZZ$ = "L" AND LL = 0 THEN
   HOME : VTAB 23: PRINT "NO S
   NIPER THERE, SIR?": FOR E =
   1 TO 1000: NEXT E: GOTO 2760
2810  IF ZZ$ = "R" AND RR = 0 THEN
   HOME : VTAB 23: PRINT "NO S
   NIPER THERE, SIR?": FOR E =
   1 TO 1000: NEXT E: GOTO 2760
2820 DR =  INT ( RND (1) * 10) +
   1: REM  DIE ROLL
2830  IF ZZ$ = "L" THEN 2860
2840  IF ZZ$ = "R" THEN 2910
2850  IF ZZ$ <  > "R" AND ZZ$ <  >
   "L" THEN 2760
2860 LDD =  INT ( SQR ((A(X,1) -
   20) ^ 2 + (A(X,2) - 35) ^ 2)
   )
2870  GOSUB 450:SH(X) = SH(X) - 1
```

```
2880  IF LDD > 100 THEN DR = DR +
      3
2890  IF DR = < 7 THEN  HOME : VTAB
      23: PRINT "GOT HIM": FOR E =
      1 TO 1000: NEXT E: HCOLOR= 0
      : DRAW 1 AT 35,35:LL = 0: GOTO
      760
2900  HOME : VTAB 23: PRINT "MISS
      ED": FOR E = 1 TO 1000: NEXT
      E: GOTO 760
2910 RDD =  INT ( SQR ((A(X,1) -
      260) ^ 2 + (A(X,2) - 35) ^ 2
      ))
2920  GOSUB 450:SH(X) = SH(X) - 1
2930  IF RDD > 100 THEN DR = DR +
      3
2940  IF DR = < 7 THEN  HOME : VTAB
      23: PRINT "GOT HIM": FOR E =
      1 TO 1000: NEXT E: HCOLOR= 0
      : DRAW 1 AT 260,35:RR = 0: GOTO
      760
2950  HOME : VTAB 23: PRINT "MISS
      ED": FOR E = 1 TO 1000: NEXT
      E: GOTO 760
2960 :
2970  REM  * GRENADE EXPLOSION *
2980 :
2990  IF EE(X) = 1 THEN 760
3000  IF GG = 0 THEN  HOME : NORMAL
      : VTAB 23: PRINT "MACHINE GU
      N ALREADY DESTROYED, SIR?": FOR
      E = 1 TO 1000: NEXT E: GOTO
      760
3010 DMG =  INT ( SQR ((A(X,1) -
      MG(1)) ^ 2 + (A(X,2) - MG(2)
      ) ^ 2))
3020  IF DMG = < RG THEN  HOME :
      VTAB 23: PRINT "NICE THROW"
      : FOR E = 1 TO 1000: NEXT E:
      GOSUB 450: XDRAW 2 AT MG(1)
      ,MG(2): HOME : VTAB 23: FLASH
      : PRINT "MACHINE GUN DESTROY
      ED!!!": FOR E = 1 TO 1000: NEXT
      E: XDRAW 2 AT MG(1),MG(2):GG
      = 0: GOTO 3670
3030 D1 = A(X,1) -  INT ((RG / DM
      G) * (A(X,1) - MG(1)))
3040 D2 = A(X,2) -  INT ((RG / DM
      G) * (A(X,2) - MG(2)))
3050 G(X) = G(X) - 1
3060  GOSUB 830
3070  GOTO 760
3080 :
3090  REM  ** INSTRUCTIONS  **
3100 :
3110  HOME : HTAB 15: PRINT "INST
      RUCTIONS": PRINT
3120  PRINT  TAB( 5)"YOUR MISSION
      IS TO LAND AND SECURE"
3130  PRINT "THE NORMANDY BEACH W
      ITH MINIMAL LOSS"
3140  PRINT "OF LIFE."
3150  PRINT : HTAB 5: PRINT "YOU
      WILL LAND 12 MEN ON THE"
3160  PRINT "BEACH AND PROCEED TO
      LOCATE AND DESTROY"
3170  PRINT "THE MACHINE GUN NEST
      LOCATED THERE."
3180  PRINT : HTAB 5: PRINT "THIS
      WILL BE A FORMIDABLE IF NOT
      "
3190  PRINT "IMPOSSIBLE TASK."
3200  PRINT : HTAB 5: PRINT "YOUR
      SOLDIERS ARE EQUIPPED ONLY"
3210  PRINT "WITH RIFLES AND 2 GR
      ENADES EACH.": PRINT
3220  HTAB 5: PRINT "HOWEVER, THE
      ENEMY IS LOCATED ON A"
3230  PRINT "CLIFF TOP OVERLOOKIN
      G THE BEACH AND HAS"
3240  PRINT "THE FOLLOWING DEFENS
      ES:-"
3250  PRINT : PRINT : INPUT "PRES
      S <RETURN> TO CONTINUE...";Z
      $
3260  IF Z$ = "" THEN 3270
3270  HOME : PRINT : HTAB 10: PRINT
      "1) A MACHINE GUN"
3280  PRINT  TAB( 10)"2) SNIPERS"
3290  PRINT  TAB( 10)"3) MORTAR S
      HELLS"
3300  PRINT  TAB( 10)"4) BARBED W
      IRE"
3310  PRINT  TAB( 10)"5) MINEFIEL
      DS"
3320  PRINT : HTAB 5: PRINT "THER
      E ARE A FEW ROCKS ON THE BEA
      CH"
3330  PRINT "WHICH WILL PROVIDE S
      OME COVER."
3340  PRINT : HTAB 5: PRINT "YOU
      MAY RUN A MAXIMUM OF 4 STEPS
      "
3350  PRINT "EACH TURN OR CRAWL A
      MAXIMUM OF 1 STEP."
3360  PRINT "YOU ARE LESS LIKELY
      TO BE HIT BY "
3370  PRINT "MACHINE GUN FIRE IF
      YOU LIE DOWN AND"
3380  PRINT "CRAWL."
3390  PRINT : HTAB 5: PRINT "YOUR
      GRENADES ARE TO BE USED ONL
      Y TO DESTROY THE MACHINE GUN
      ."
3400  PRINT : PRINT : INPUT "PRES
      S <RETURN> TO CONTINUE...";Z
      $
3410  IF Z$ = "" THEN 3420
3420  HOME : PRINT : HTAB 5: PRINT
      "THE COMMANDS TO MOVE YOUR M
      EN"
3430  PRINT "ARE AS FOLLOWS:-"
3440  PRINT : PRINT : HTAB 10: PRINT
      "R-RUN"
3450  PRINT  TAB( 10)"C-CRAWL"
3460  PRINT : PRINT "THEN PRESS T
      HE FOLLOWING KEYS:-": PRINT
3470  PRINT  TAB( 10)"'I'-TO MOVE
      UP"
3480  PRINT  TAB( 10)"'J'-TO MOVE
      LEFT"
3490  PRINT  TAB( 10)"'K'-TO MOVE
      RIGHT"
3500  PRINT  TAB( 10)"'M'-TO MOVE
      DOWN"
3510  PRINT  TAB( 10)"'S'-TO SET
      MAN IN NEW POSITION"
3520  PRINT : PRINT : INPUT "PRES
      S <RETURN> TO CONTINUE...";Z
      $
3530  IF Z$ = "" THEN 3540
3540  HOME : VTAB 3: PRINT  TAB(
      7)"OTHER COMMANDS ARE:-"
3550  PRINT : PRINT  TAB( 7)"D-LI
      E DOWN"
3560  PRINT  TAB( 7)"U-STAND UP"
3570  PRINT  TAB( 7)"A-AIM RIFLE"
3580  PRINT  TAB( 7)"F-FIRE RIFLE
      "
3590  PRINT  TAB( 7)"L-LOAD RIFLE
      "
3600  PRINT  TAB( 7)"P-PULL PIN (
      OF GRENADE)"
3610  PRINT  TAB( 7)"T-THROW GREN
      ADE (AT MACHINE GUN)"
3620  PRINT  TAB( 7)"?-REFER TO C
      OMMAND LIST"
3630  PRINT  TAB( 7)"Q-QUIT PROGR
      AM"
3640  VTAB 16: PRINT  TAB( 7)"GOO
      D LUCK!-- YOU'LL NEED IT!!!"
3650  VTAB 20: PRINT : INPUT "PRE
      SS <RETURN> TO BEGIN...";Z$
3660  GOTO 200
3670 :
3680  REM  ** END **
3690 :
3700  TEXT : NORMAL : HOME : HTAB
      10: VTAB 10
3710  PRINT "CONGRATULATIONS!!": PRINT
3720  PRINT "YOU HAVE COMPLETED Y
      OUR MISSION"
3730  PRINT "SUCCESSFULLY."
3740  FOR X = 1 TO 12
3750  IF ST$(X) = "KILLED" THEN N
      K = NK + 1
3760  NEXT X
3770  PRINT : PRINT "NO. OF MEN K
      ILLED=";NK"
3780  PRINT : PRINT : INPUT "CARE
      TO TRY AGAIN, CHAMP? (Y/N)
      ";ZZ$
3790  IF ZZ$ = "Y" THEN  HOME : GOTO
      160
3800  IF ZZ$ <  > "Y" AND ZZ$ <  >
      "N" THEN 3780
3810  END
3820  NORMAL : TEXT : HOME
3830  HTAB 10: VTAB 10: PRINT "GA
      ME OVER."
3840  PRINT : INPUT "CARE TO PLAY
      AGAIN? (Y/N) ";ZZ$
3850  IF ZZ$ = "Y" THEN  HOME : GOTO
      160
3860  IF ZZ$ <  > "Y" AND ZZ$ <  >
      "N" THEN 3820
3870  HOME : HTAB 10: VTAB 10: PRINT
      "BYE BYE": END
```

```
* NORMANDY  BEACH *

0800- 00 02 00 06 00 1D 00 06
0808- 07 05 05 05 05 07 07 07
0810- 06 06 07 06 07 05 04 05
0818- 05 06 05 07 00 00 05 06
0820- 07 06 05 06 07 06 07 07
0828- 04 07 04 07 04 06 05 06
0830- 05 06 05 07 05 05 05 04
0838- 05 04 05 07 06 07 06 07
0840- 06 07 07 07 06 07 07 07
0848- 07 07 07 05 05 06 06 07
0850- 06 04 05 04 05 05 04 05
0858- 05 05 05 05 05 05 04 05
0860- 04 05 04 06 07 06 06 05
0868- 07 07 07 07 07 06 06 05
0870- 06 05 06 05 06 04 07 04
0878- 07 07 06 06 06 06 07 04
0880- 04 04 04 04 04 07 06 07
0888- 06 07 06 07 06 07 05 00
0890- 06 EF
```

擊手被射殺的可能性。

行句2970—3060是手榴彈爆炸部份。

行句3010決定士兵與大砲的距離，即DMG。見圖(3)。在行句3020中，假如離開DMG距離相等或少於手榴彈投擲的距離（RG），大砲便被擊中。圖(4)顯示出爆炸方位的計算辦法。

至此，大家應該非常清楚了解整個程式的結構及編寫方法。以後在編寫這類程式時，大家不妨參考。同時筆者還將程式中所用到之變數及旗號詳列於表(1)。

各位有興趣的讀者可自行將程式改良。以下筆者提供幾個範圍供大家參考之用。

(1) 將士兵移動步數改變。行句1770的S.T＝5或行句1780的ST＝20。

(2) 將追擊砲的數目增加，見行句2160。

(3) 將大砲的反應改變。行句2470中的15或2480中的9，或將2490的DR值改變以便在伏地狀態是不易被流彈擊中。2500至2520中士兵的Y位置值亦可改變。

(4) 改變行句2170中的追擊砲彈命中率。

(5) 改變行句360 中的手榴彈爆炸範圍。

(6) 將狙擊手的射擊命中率改變（行句2630—2690）。

(7) 重置地雷陣的位置（220至300）。

除了這些改變外，大家可以試將遊戲的刺激性再提高例如增加多些狙擊手的數目，提高敵方的還擊力量。又例如，當螢幕上所有狙擊手皆陣亡後。

假如你有能力更可將整個程式重新編排，將遊戲改為兩人玩，一個是攻擊，一個是防守。

最後，大家又可以增加大砲的數目，使遊戲難度更加提高。

本文程式有兩個檔案，分別為NORMANDY WAR及NORMANDY BEACH。■

不能LIST的秘密



解解拆拆

各位讀者曾否試過LIST 一個程式出來時，只看到程式的第一行呢？或者看見不斷列出程式第一行的情況呢？前者是用POKE 2049，0 所造成的，而後者則是用POKE 2049，1產生的。

其實，BASIC程式是以$801（2049）開始儲存的，而第一個BIT是指出程式下一行的位置，第二BIT是表示行數。因此如果我們將$801 （2049）之值改為0，則將第一行指標指向$800，而$800 的值是$00，故電腦會誤認程式已結束而停止列印。另一方面，如我們將$801 之值改為1，則將第一行指標指向$801 ，因此電腦便會不斷地列出第一行。

至於解決方法很簡單。但首先我們要知道當每一行程式結束時便會以$00代表（在監督程式下便可見），接着便是下一行的行數。要列出全部程式，只要CALL—151 進入監督程式，鍵入800，8xx ，列出BASIC開頭一段程式（長短隨能列出之句的長短而定）。找出第二句程式之儲存位址，然後將之放入$801 即可。

為使各位讀者更加明白，現試舉一例說明之。

首先鍵入下列程式：

```
5   REM  TEST PROGRAM
10  HOME : HTAB 5: VTAB 7: PRINT
     "THIS PROGRAM"
15  HTAB 3: VTAB 8: PRINT "IS USE
     D TO TEST 'POKE 2049,0'"
20  HTAB 3: VTAB 9: PRINT "TYPE P
     OKE 2049,0,AFTER YOU TYPE TH
     IS PROGRAM"
25  END
```

再打POKE 2049，0

執行無誤後，（結果應是只能列出LINE 5）

鍵入CALL—151 ，*800，820

（由於LINE 5 不是很長，故只大約列出800，820 的值便足夠了）如下：

```
                      *800.820
第一行之LINE NO. ——
                      0800- 00 00 08 [05 00] B2 20 54
                      0808- 45 53 54 20 50 52 4F 47
                      0810- 52 41 4D 00 [30 08] 0A 00
下一行程式之儲存地址 ——
                      0818- 97 3A 96 35 3A A2 37 3A
                      0820- BA
```

因此，只要把放"30"位置的相對地址，即$814 放入801 及802，但由於大家都是同在800～900 之間的，故只須將14放入801 即可!!

即，鍵入*801：14

再CTRL—C 回到BASIC，打入LIST。

怎樣，成功了吧！現在你能依照上列步驟去LIST出其他用POKE 2049，0保護的程式了。

LOCKSMITH 5.0

TEXT EDITOR 的使用方法

LOCKSMITH 5.0 自從推出以來，曾經推出過一批新的參數，在它那一百四十頁厚的MANUAL 中由第93頁起曾對LPL（LOCKSMITH PROGRAMMING LANGUAGE）的語法的每個命令的格式及用途解釋過，不過它却不是甚麼參數，在觀察之下它是有些和ＬＯＧＯ相似的語言，它比較採用數字改換參數的舊方法要好多了，最起碼它容易使人明白，易於掌握，在它今年３月份發表的NEWSLETTER之中，曾說明ＬＰＬ的使用及推出一批新的參數，使我們可以對各種不同的程式進行ＣＯＰＹ，由於這些參數的數量不少，因此它們被貯於磁碟之中待需要時取出使用（這和NIBBLE AWAY的AUTOLOAD FILE 很相似），它們都可以存入一個大磁碟檔中（卽ＬＰＬ FILES）以節省磁碟空間，當然這比單一存放要經濟得多，但想從此大磁碟檔中載入某段參數，方法和載入整個磁碟檔有些不同，以下我會和各位研究一下這個問題。

```
HEX 0000000000000000111111111111111112222
TRK 0123456789ABCDEF0123456789ABCDEF0123
.00
.25
.50
.75
                 TEXT EDITOR

*** DEFAULT LOCKSMITH LPL FILE
 READ
 CHANGE SS DHDR
 TSTART DOS PAT1 STAT 1
 IF FAIL 1 THEN TSTART LONG SS
 TEND REPEAT
 DISPLAY
 TSTART BSPACE
 TEND FSPACE 3
 VSTART NORM
 LABEL 1:WRITE STAT 1
 IF SUCC 1 THEN GOTO EE
 SHORTEN ALL EQUAL BY 2 STAT 5
 IF FAIL 5 THEN GOTO DD
 CLEAR ERROR : GOTO 1
 LABEL DD
 TEND FSPACE 3
 VSTART NORM
 LABEL 1:WRITE STAT 1
 IF SUCC 1 THEN GOTO EE
 SHORTEN ALL EQUAL BY 2 STAT 5
 IF FAIL 5 THEN GOTO DD
 CLEAR ERROR : GOTO 1
 LABEL DD
 SHORTEN ALL CENTER BY 2 STAT 6
 IF FAIL 6 GOTO DA
 CLEAR ERROR : GOTO 1
 LABEL DA: * SHORTENING FAILED
 LABEL EE: * WRITE SUCCEEDED
**
```

首先，我們來看一下一個程式磁片的參數在磁碟檔中存放的情形，下面我就以 DEFAULT LOCKSMITH LPL FILE 來作一說明：（圖１）

在圖一的參數檔中，有兩件很重要的事要注意，①在每個參數列表的最前面含有兩個星號（**），而後面卽是跟著參數本身的檔名（本例是大家的LOCKSMITH 磁碟中都存有的DEFAULT LPL FILE），而②在參數列表的最後也有三個星號（***），上面所提列的兩個及三個星號，主要是作爲一個參數檔的開頭和結尾的識別信號，因爲有這些識別信號，所以在LPL FILE之中能夠存入許多不同的程式參數檔。

由LPL FILE中載入部份參數的方法

正如我們上面所說，在LPL FILE中，同時存有很多不同的程式COPY參數，若要由LPL FILE中載入某一程式的COPY參數，其步驟如下：

首先從MAIN MENU 中鍵入Ｔ選擇TEXT EDITOR功能，當TEXT EDITOR從LOCKSMITH 碟中載入後，螢幕上會出現EDITOR MENU ，此時可鍵入Ｌ，以選擇載入參數的功能，接著你會被要求輸入檔名，LOCKSMITH 5.0之參數是以往COPY 磁片第一個英文單字來歸檔的，如上例的DEFAULT之〝D〞，卽PARMS D檔，另外，我們假如要COPY的是APPLE WRITER，則是PARMS Ａ檔。

當我們將所要參數之檔名（例如PARMS D或PARMS A）輸入後，接著再按下RETURN 鍵，此時你會被要求輸入要找尋之字串。關於這一點，各位在看完以下說明後便會明白。

搜尋字串（卽是所欲BACK／UP 之磁片名稱）是以下面的方式進行工作的，例如在上述的例子中，我們應輸入的搜尋字串是〝DEFAULT LOCKSMITH LPL FILE〞（或APPLE WRITER）。在輸入欲搜尋之字串後，LOCKSMITH卽會開始在整個檔案中，找尋兩個星號（**）之後，是否含有所指定之字串（注意：在**與字串之間無SPACE 插入）。

在LOCKSMITH 找到指定之字串（卽所要載入參數之起點）後，卽開始進行參數載入動作，直到發現三個星號（***），代表參數之結束，才停止載入動作，且又將控制權交回給TEXT EDITOR ，此時你卽可鍵入Ｂ後，放入一LOCKSMITH 5.0 磁碟以執行BACK UP 的工作，或利用EDITOR所提供的其它功能，例如鍵入Ｅ，可以查看參數是否所需要的，如果是空白的話，則是無此參數，另外也可在這功能中修改參數。

上面所提到的功能相當有用，但是操作時必須小心，以便使進行正確的COPY 。■

GRAPHICS MAGAICIAN編寫的遊戲：

聲音和畫面的技巧

麥法基

這一期讓我們研究APPLE 的聲音和畫面問題。在各種不同電腦中，APPLE 的聲音無疑是比較弱一環。聲音產生由＄C030 的輸出位址來觸動喇叭，倘若觸動次數多的話，它便會產生頻率高的音調。換言之，所有不同聲音或音調都需要程式設計者去創造出來。由於只有一個位址，所以無論音響或音樂都比較單純，有別於其他電腦例如ATARI或COMMODORE。該等電腦音響部份都有獨立的晶片負責，所以能造出豐富的音色和音響效果。

雖然APPLE 的發聲部份有限制，但仍然可以由程式來達到一定的效果。最簡單的聲音莫如一些單純的機器聲音，我們可以用P＝PEEK （—16336）或機械語STA＄C030來產生，然後重覆以上指令或在程式內加以迴路便可。假如需要純音樂的樂音，亦可以利用程式控制喇叭以觸動次數而達到目的。

LISTING #1 - SPEAKER ROUTINE

```
1000  REM  SPEAKER ROUTINE
1010  DATA  160,0,174,0,3,173,48,
       192
1020  DATA  136,208,5,206,1,3,240
       5
1030  DATA  202,208,245,240,237,9
       6,256
1040 AD = 770: REM  PUT INTO ADDR
       ESS $302
1050  READ X
1060  IF X = 256 THEN  RETURN
1070  POKE AD,X
1080 AD = AD + 1
1090  GOTO 1050
2000  REM  PITCH=P:TIME=T
2010  POKE 768,P: POKE 769,T
2020  CALL 770: RETURN
```

參考LISTING1 的SPEAKER ROUTINE。這是一個最常用的觸動喇叭程式，無須詳細在這裏解釋，請鍵入以留後用。因爲無論聲音怎樣，都需要這個程式產生音響。

舉例來說如果我們想產生一種激光發射的音響效果，怎樣利用它來加入程式內呢？原則上，激光發射的聲響是把聲音的音調在最短時間內由最高降至最低。請在LISTING 1 的程式內加上第10至140 行變成LISTING 2 的程式然後執行。

螢幕首先顯示CYCLE？字樣，問我們需要激光音響的次數。在我們來說，暫時一次便足夠了，所以鍵入1。繼續電腦會詢問聲音開始時的音調STARTING PITCH，數值越少則音調會

LISTING #2 - SOUND DEMO

```
10  REM  SOUND DEMO
20  TEXT : HOME : GOSUB 1000
30  INPUT "NO. OF CYCLE?";NC
40  INPUT "STARTING PITCH?";P1
50  INPUT "ENDING PITCH?";P2
60  INPUT "PITCH INCREMENT? (MINU
     S SIGN FOR DECREMENT)";PI
70  INPUT "TIME OF SOUND?";T
80  FOR J = 1 TO NC
90  FOR P = P1 TO P2 STEP PI
100  GOSUB 2000: NEXT P
110  NEXT J
120  PRINT "DO YOU WANT TO HEAR A
      GAIN? (Y/N)"
130  GET A$: IF A$ <  > "Y" THEN
      30
140  GOTO 80
1000  REM  SPEAKER ROUTINE
1010  DATA  160,0,174,0,3,173,48,
       192
1020  DATA  136,208,5,206,1,3,240
       5
1030  DATA  202,208,245,240,237,9
       6,256
1040  AD = 770: REM  PUT INTO ADDR
       ESS $302
1050  READ X
1060  IF X = 256 THEN  RETURN
1070  POKE AD,X
1080  AD = AD + 1
1090  GOTO 1050
2000  REM  PITCH=P:TIME=T
2010  POKE 768,P: POKE 769,T
2020  CALL 770: RETURN
```

LISTING #3 - ALIEN.SSH

```
6300.6403

6300- 00 0E 00 31 00 54 00 77
6308- 00 9A 00 BD 00 E0 03 0B
6310- 00 00 00 00 00 00 50 02
6318- 00 54 0A 00 34 0B 00 54
6320- 0A 00 50 02 00 54 0A 00
6328- 44 08 00 00 00 00 00 00
6330- 00 03 0B 00 00 00 00 00
6338- 00 20 05 00 28 15 00 68
6340- 16 00 28 15 00 20 05 00
6348- 28 15 00 08 11 00 00 00
6350- 00 00 00 00 03 0B 00 00
6358- 00 00 00 00 40 0A 00 50
6360- 2A 00 50 2D 00 50 2A 00
6368- 40 0A 00 50 2A 00 40 08
6370- 00 00 00 00 00 00 00 03
6378- 0B 00 00 00 00 00 00 00
6380- 15 00 20 55 00 20 5B 00
6388- 20 55 00 00 15 00 20 55
6390- 00 00 11 00 00 00 00 00
6398- 00 00 03 0B 00 00 00 00
63A0- 00 00 00 2A 00 40 2A 01
63A8- 40 36 01 40 2A 01 00 2A
63B0- 00 40 2A 01 40 08 01 00
63B8- 00 00 00 00 00 03 0B 00
63C0- 00 00 00 00 00 00 54 00
63C8- 00 55 02 00 6D 02 00 55
63D0- 02 00 54 00 00 55 02 00
63D8- 11 02 00 00 00 00 00 00
63E0- 03 0B 00 00 00 00 00 00
63E8- 00 28 01 00 2A 05 00 5A
63F0- 05 00 2A 05 00 28 01 00
63F8- 2A 05 00 08 01 00 00 00
6400- 00 00 00 55
```

LISTING #4 - TANK.SSH

```
6300- 00 0E 00 34 00 5A 00 80
6308- 00 A6 00 CC 00 F2 04 09
6310- 00 84 00 00 00 95 00 00
6318- 00 95 00 00 00 95 00 00
6320- 00 8E 00 00 90 8E 81 00
6328- F4 FF 85 00 F4 FF 85 00
6330- 94 8E 85 00 04 09 00 88
6338- 00 00 00 AA 00 00 00 AA
6340- 00 00 00 AA 00 00 00 9C
6348- 00 00 A0 9C 82 00 E8 FF
6350- 8B 00 E8 FF 8B 00 A8 9C
6358- 8A 00 04 09 00 90 00 00
6360- 00 D4 00 00 00 D4 00 00
6368- 00 D4 00 00 00 B8 00 00
6370- C0 B8 84 00 D0 FF 97 00
6378- D0 FF 97 00 D0 B8 94 00
6380- 04 09 00 A0 00 00 00 A8
6388- 81 00 00 A8 81 00 00 A8
6390- 81 00 00 F0 80 00 00 F1
6398- 88 00 A0 FF AF 00 A0 FF
63A0- AF 00 A0 F1 A8 00 04 09
63A8- 00 C0 00 00 00 D0 82 00
63B0- 00 D0 82 00 00 D0 82 00
63B8- 00 E0 81 00 00 E2 91 00
63C0- C0 FE DF 00 C0 FE DF 00
63C8- C0 E2 D1 00 04 09 00 00
63D0- 81 00 00 A0 85 00 00 A0
63D8- 85 00 00 A0 85 00 00 C0
63E0- 83 00 00 C4 A3 00 00 FD
63E8- BF 81 00 FD BF 81 00 C5
63F0- A3 81 04 09 00 00 82 00
63F8- 00 C0 8A 00 00 C0 8A 00
6400- 00 C0 8A 00 00 00 87 00
6408- 00 88 C7 00 00 FA FF 82
6410- 00 FA FF 82 00 8A C7 82
6418- FF
```

越高，鍵入40這個數值比較適合。（音調太尖效果並不合符理想）跟著是聲音完結時音調，鍵入140 然後螢幕再詢問PITCH INCREMENT？（MINUS SIGN FOR DECREMENT）。雖然聲調由高至低，但我們亦可以把升降時的每個音程距離加以控制，使產生不同的效果。鍵入8看看效果如何？（負數是降低的數值，即音調由最低至最高，請記着數值越大，音調越低。）

最後是聲音產生（每個聲調）的時間，太長時間會失去激光效果，而且當我們把聲音加進遊戲程式內運行，會減慢畫面運行速度，所以鍵入４便可。所有資料鍵入後，APPLE 的喇叭便會立即發出激光發射的音响效果了，很簡單是不是？請繼續鍵入不同數值看看其他音响效果如何，然後選擇更好的聲音。

你或許會發覺到執行以上程式時，音响有些少斷續的感覺。原因是由BA—SIC 執行，如果由機械語言執行就更加完美了，以下會談到。現在請以LIS—TING ＃3至LISTING ＃7 的資料造一個DEMO. ANM 的動畫程式。—

LISTING #5 - BULLET.SSH

```
6300- 00 0E 00 20 00 32 00 44
6308- 00 56 00 68 00 7A 02 08
6310- 03 00 03 00 03 00 03 00
6318- 03 00 00 00 00 00 00 00
6320- 02 08 06 00 06 00 06 00
6328- 06 00 06 00 00 00 00 00
6330- 00 00 02 08 0C 00 0C 00
6338- 0C 00 0C 00 0C 00 00 00
6340- 00 00 00 00 02 08 18 00
6348- 18 00 18 00 18 00 18 00
6350- 00 00 00 00 00 00 02 08
6358- 30 00 30 00 30 00 30 00
6360- 30 00 00 00 00 00 00 00
6368- 02 08 60 00 60 00 60 00
6370- 60 00 60 00 00 00 00 00
6378- 00 00 02 08 40 01 40 01
6380- 40 01 40 01 40 01 00 00
6388- 00 00 00 00 FF
```

LISTING #6 - ALIEN PATH.PTH

```
4000- 08 08 08 08 08 28 28 28
4008- 28 28 28 08 08 08 08 08
4010- 08 08 08 08 28 28 28 28
4018- 28 08 08 08 08 08 08 08
4020- 08 08 08 08 08 08 08 08
4028- 88 88 88 88 08 08 08 08
4030- 08 08 08 08 08 08 88 88
4038- 88 88 08 08 08 08 08 08
4040- 08 08 08 08 88 88 80 80
4048- 80 82 82 82 82 02 02 02
4050- 02 02 02 02 02 82 82 82
4058- 02 02 02 02 02 02 02 02
4060- 02 02 22 22 22 22 02 02
4068- 02 02 02 02 02 82 82 82
4070- 82 02 02 02 02 02 02 02
4078- 02 02 02 02 02 02 02 02
4080- 02 02 02 02 02 22 22 22
4088- 02 02 02 02 02 02 02 22
4090- 22 22 28 28 28 00 00
```

LISTING #7 - BULLET PATH.PTH

```
4000- C0 00 08
```

LISTING #9 - DEMO.SOURCE

```
**END OF PASS 1

**END OF PASS 2

4000                      ORG $4000
4000                      OBJ $800
4000 4C 05 40             JMP BEGIN
4003           ;
4003           ; ******************************
4003           ; *                            *
4003           ; * BSAVE THE OBJ. FILE WITH   *
4003           ; * "DEMO.OBJ" AFTER SOURCE    *
4003           ; * CODE ASSEMBLED.            *
4003           ; *                            *
4003           ; ******************************
4003           ;
4003           ;
4003           ;
0000           XBYTE     EPZ $0
0001           XBIT      EPZ $1
0002           VERT      EPZ $2
0003           SHNUM     EPZ $3
4003           ;
4003           ;
94A0           AXBYTE    EQU $94A0          ; PEEK(38048)
LISA 2.5

94A1           AXBIT     EQU $94A1
94A2           AVERT     EQU $94A2          ; PEEK(38049)
94A3           BXBYTE    EQU $94A3          ; PEEK(38050)
94A4           BXBIT     EQU $94A4          ; PEEK(38051)
94A5           BVERT     EQU $94A5          ; PEEK(38052)
C061           BUTTON    EQU $C061          ; PEEK(38053)
FCA8           DELAY     EQU $FCA8
0300           PITCH     EQU $300
0301           TIME      EQU $301
0302           SOUND     EQU $302
4003           ;
4003           ;
4003           BFLAG     DFS 1
4004           TEMP      DFS 1
4005           ;
4005           ;
4005           ;
4005           ;STARTING LINE 50 OF LISTING #8
4005           ;
4005 A9 00     BEGIN     LDA #$00
4007 8D 03 40            STA BFLAG
400A AD A5 94  START     LDA BVERT          ; BFLAG=0
400D C9 E6               CMP #$E6
400F B0 03               BGE START.1        ;IF PEEK(38053)>230
4011 4C 25 40            JMP START.3
4014 C9 F0     START.1   CMP #$F0
4016 90 03               BLT START.2        ; AND PEEK(38053)<240
4018 4C 25 40            JMP START.3
401B A5 FE     START.2   LDA $FE
401D 8D 43 94            STA $9443
4020 A9 00               LDA #$0            ;POKE 37955,254
4022 8D 03 40            STA BFLAG
4025 AD 03 40  START.3   LDA BFLAG          ; BFLAG=0
                                            ;IF BFLAG=1 THEN
```

LISTING #8 - DEMO.1

```
10  HGR : POKE  - 16302,0:T = 3: GOSUB
     1000
20  PRINT  CHR$ (4);"BLOAD DEMO.A
     NM"
30  POKE 0,20: POKE 1,0: POKE 2,1
     50: POKE 3,2: CALL 36608
40  POKE 37955,254
50 BFLAG = 0
60  IF  PEEK (38053) > 230 AND  PEEK
     (38053) < 240 THEN  POKE 379
     55,254:BFLAG = 0
65  IF BFLAG = 1 THEN 80
70  IF  PEEK ( - 16287) > 127 THEN
      POKE 38051,21: POKE 38052,0
      : POKE 38053,146: POKE 37955
      ,0:BFLAG = 1: GOSUB 900
80  CALL 36928
90  GOTO 60
900  FOR P = 40 TO 140 STEP 8
910  GOSUB 2000: NEXT P
920  RETURN
1000  REM  SPEAKER ROUTINE
1010  DATA  160,0,174,0,3,173,48,
      192
1020  DATA  136,208,5,206,1,3,240
      ,5
1030  DATA  202,208,245,240,237,9
      6,256
1040  AD = 770: REM  PUT INTO ADDR
      ESS $302
1050  READ X
1060  IF X = 256 THEN  RETURN
1070  POKE AD,X
1080  AD = AD + 1
1090  GOTO 1050
2000  REM  PITCH=P:TIME=T
2010  POKE 768,P: POKE 769,T
2020  CALL 770: RETURN
```

```
]
DEMO.ANM - ENTRIES TABLE

OBJECT NUMBER : 0

SHAPE : 0

START AT 200, 40

PATHS :
0) 0
1) REPEAT
2)
3)

STARTING PATH : 0
LOCATION IN PATH : 0

OBJECT NUMBER : 1

SHAPE : 1

START AT 140, 192

PATHS :
0) 1
1) REPEAT
2)
3)

STARTING PATH : 0
LOCATION IN PATH : 0
```

REMARKS:-USE BLOCK AND PAGE 1 IN ANIMATION ROUTINE

LISTING #8A - DEMO.2

```
10  HGR : POKE  - 16302,0:T = 3: GOSUB
     1000
20  PRINT  CHR$ (4);"BLOAD DEMO.A
     NM"
30  POKE 0,20: POKE 1,0: POKE 2,1
     50: POKE 3,2: CALL 36608
40  POKE 37955,254
50 BFLAG = 0
60  IF  PEEK (38053) > 230 AND  PEEK
     (38053) < 240 THEN  POKE 379
     55,254:BFLAG = 0
65  IF BFLAG = 1 THEN P = P + 8: GOSUB
     900: GOTO 80
70  IF  PEEK ( - 16287) > 127 THEN
      POKE 38051,21: POKE 38052,0
      : POKE 38053,146: POKE 37955
      ,0:BFLAG = 1:P = 40: GOSUB 9
      00
80  CALL 36928
90  GOTO 60
900  IF P < 148 THEN  GOSUB 2000
910  RETURN
920  RETURN
1000  REM  SPEAKER ROUTINE
1010  DATA  160,0,174,0,3,173,48,
      192
1020  DATA  136,208,5,206,1,3,240
      ,5
1030  DATA  202,208,245,240,237,9
      6,256
1040  AD = 770: REM  PUT INTO ADDR
      ESS $302
1050  READ X
1060  IF X = 256 THEN  RETURN
1070  POKE AD,X
1080  AD = AD + 1
1090  GOTO 1050
2000  REM  PITCH=P:TIME=T
2010  POKE 768,P: POKE 769,T
2020  CALL 770: RETURN
```

SHAPE NO.0 是ALIEN, SSH，SHA—PE NO.1 是BULLET、SSH，SHA—PE NO.2 是TAK、SSH，PATH NO.0是ALIEN PATH而PATH NO.1是BULLET PATH 。讀者只要根據TABLE #1 的資料輸入即可。

讓我們來看看把聲音加入畫面內運行的效果如何，把LISTING #2 改成LISTING #8 然後執行。畫面上有一隻遊划的怪物和一艘太空船，你只要按BUTTON 0便會有子彈發射而且配有先前的激光音响。這是一個動作很慢的動畫，而且當激光發射時畫面動作會暫停讓聲音產生。這不是我們想需要的效果吧！畫面動作慢大家都知道是基於—BASIC 語言的影响，但我們總不能讓畫面停下來讓路給聲音運行。問題在於第900行的聲音程式，聲音由高至低的迴路影响。於是我們把整個程式的第65行和900 行的程式更改成爲LISTING #8A 然後再執行。現在聲音和畫面不是同時進行了嗎？原因相信不用我多加解釋了。千萬不要認爲BASIC 是一個緩慢的語言而把它忽略，它是機械語言的一個基礎。

知道聲音的分割基礎後，我們把以上的程式改寫爲機械語言。LISTING #9是組合語言的原始碼，把它鍵入然後組合起來存放在磁碟上，再鍵入LIS—TING #10的程式執行。現在無論聲音畫面都十分完美地在你面前運行了。LISTING #10是LISTING #8A 改爲交給機械語言運行的程式，從LISTING #9 的注解欄可以大約知道根據BASIC 轉爲MACHINE CODE 的指令。由於機械語太快，所以加入第88行的延遲，至於第98行的WAIT是SPEAKER 的翻版，只不過並不觸動喇叭。如此當子彈發聲完畢仍然以同一速度移動，不會突然變得快起來。

```
4028 F0 2B      55          BEQ START.4
402A AD 00 03   56          LDA PITCH
402D C9 8C      57          CMP ##$8C
402F B0 1E      58          BGE START.3A
4031 EE 00 03   59          INC PITCH
4034 EE 00 03   60          INC PITCH
4037 EE 00 03   61          INC PITCH
403A EE 00 03   62          INC PITCH
403D EE 00 03   63          INC PITCH
4040 EE 00 03   64          INC PITCH
4043 EE 00 03   65          INC PITCH
4046 EE 00 03   66          INC PITCH           ;P=P+8
4049 20 86 40   67          JSR LASER           ;GOSUB 900 OF BASIC
404C 4C 7B 40   68          JMP LOOP
404F 20 8F 40   69  START.3A JSR WAIT
4052 4C 7B 40   70          JMP LOOP
4055 AD 61 C0   71  START.4 LDA BUTTON          ;IF PEEK(-16287)
4058 10 21      72          BPL LOOP
405A A9 15      73          LDA ##$15           ;>127 THEN
405C 8D A3 94   74          STA BXBYTE          ;POKE 38051,21
405F A9 00      75          LDA ##$00
4061 8D A4 94   76          STA BXBIT           ;POKE 38052,0
4064 A9 92      77          LDA ##$92
4066 8D A5 94   78          STA BVERT           ;POKE 38053,146
4069 A9 00      79          LDA ##$00

LISA 2.5

406B 8D 43 94   80          STA $9443           ;POKE 37955,0
406E A9 01      81          LDA ##$1
4070 8D 03 40   82          STA BFLAG           ;BFLAG=1
4073 A9 28      83          LDA ##$28
4075 8D 00 03   84          STA PITCH           ;P=40
4078 20 86 40   85          JSR LASER           ;GOSUB 900 OF BASIC
407B 20 40 90   86  LOOP    JSR $9040           ;CALL 36928
407E A9 40      87          LDA ##$40
4080 20 A8 FC   88          JSR DELAY           ; SMALL DELAY
4083 4C 0A 40   89          JMP START           ;GOTO 60 OF BASIC
4086            90  ;
4086            91  ;
4086 A9 03      92  LASER   LDA ##$3
4088 8D 01 03   93          STA TIME
408B 20 02 03   94          JSR SOUND           ;GOSUB 2000 OF BASIC
408E 60         95  LASER.1 RTS
408F            96  ;
408F            97  ;
408F A9 03      98  WAIT    LDA ##$3
4091 8D 01 03   99          STA TIME
4094 AD 04 40   100 WAIT1   LDA TEMP
4097 88         101 S1      DEY
4098 D0 05      102         BNE S2
409A CE 01 03   103         DEC TIME
409D F0 09      104         BEQ SEND
409F CA         105 S2      DEX
40A0 D0 F5      106         BNE S1
40A2 AE 00 03   107         LDX PITCH
40A5 4C 94 40   108         JMP WAIT1
40A8 60         109 SEND    RTS
40A9            110 ;
40A9            111 ;
40A9            112         END

***** END OF ASSEMBLY
```

ZYTRO WAR MOCKINGBOARD版本

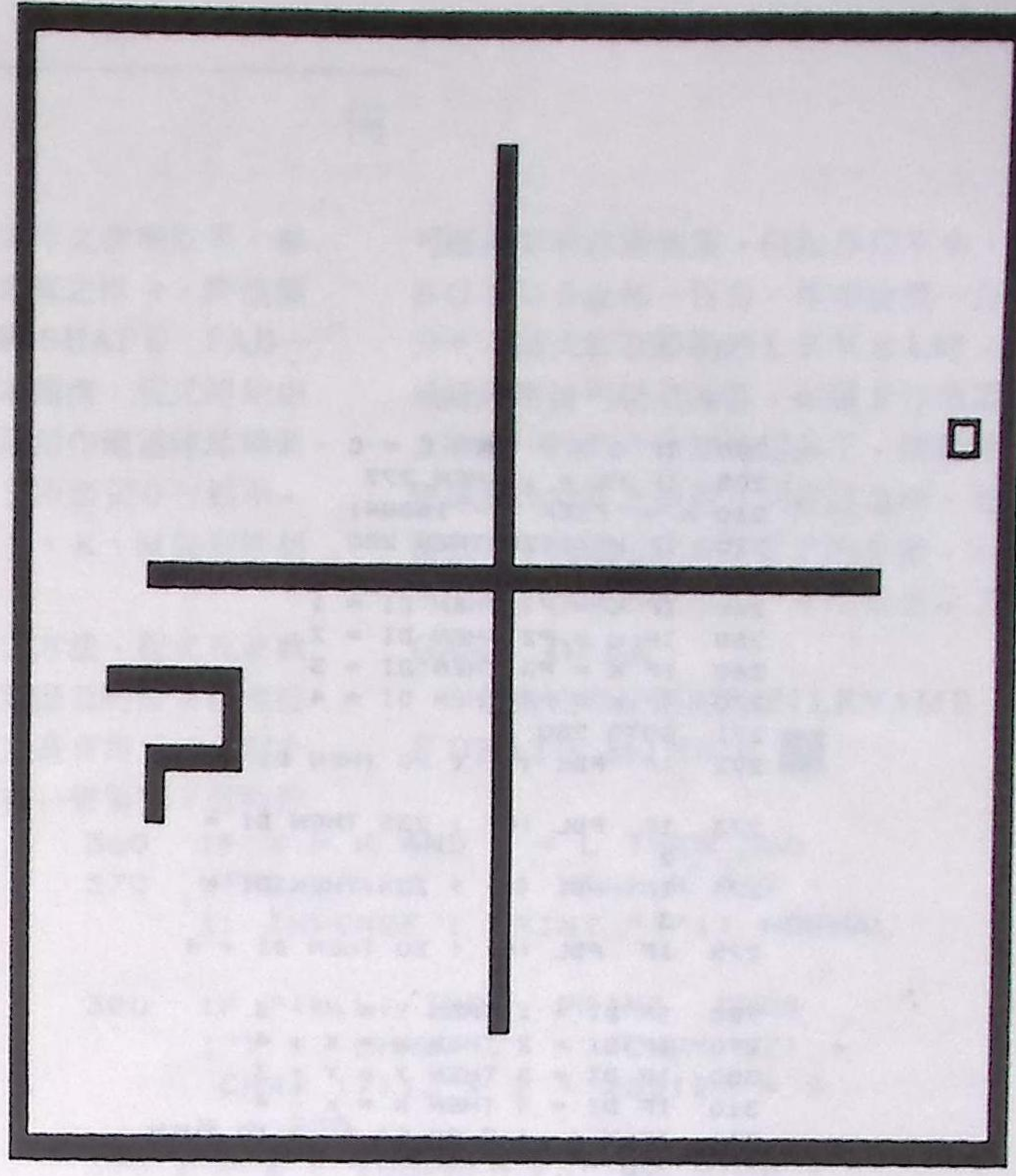
雖然APPLE 的聲音有所限制，不過最近市面出售一張MOCKING BOA—R D 的介面卡，却可以把APPLE 加上十分豐富的音响效果，而且是立體聲每邊包括三個聲度。筆者把ZYTRO WAR配上MOCKING BOARD 後發覺整個遊戲有脫胎換骨之感，大大增加了現場感受。讀者有興趣可到電腦時代詢問，下次介紹另外一個遊戲，再會。■

LISTING #10 - DEMO.3

```
10  HGR : POKE - 16302,0: GOSUB
     1000
15  PRINT CHR$ (4);"BLOAD DEMO.O
     B,A$4000"
20  PRINT CHR$ (4);"BLOAD DEMO.A
     NM"
30  POKE 0,20: POKE 1,0: POKE 2,1
     50: POKE 3,2: CALL 36608
40  POKE 37955,254
50  CALL 16384
1000  REM  SPEAKER ROUTINE
1010  DATA  160,0,174,0,3,173,48,
     192
1020  DATA  136,208,5,206,1,3,240
     ,5
1030  DATA  202,208,245,240,237,9
     6,256
1040 AD = 770: REM  PUT INTO ADDR
     ESS $302
1050  READ X
1060  IF X = 256 THEN  RETURN
1070  POKE AD,X
1080 AD = AD + 1
1090  GOTO 1050
```

新改良蛇之噩夢

Lee Wai Kwok



這個程式和以前的程式不同了。就是你要在一定時間內吃到一個蘋果，否則便GAME OVER。當你吃夠十個蘋果後，你便可以看到有一個出口，你只要逃走後，便會升上一個ＬＥＶＥＬ。

按ＲＵＮ後，電腦便問你用 JOYSTICK或KEYBO—ARD 玩，然後再問你玩那一個ＬＥＶＥＬ（１）—容易（２）—較難（３）—最難。若是使用KEYBOARD 的話便再問你是否想用其他鍵控制。

行號8,500—570 是用作改掉其他鍵控制。（可以不鍵入）

行號175 是用作發出聲音的。（可以不鍵入）

整個程式都沒有什麼特別的地方，不過都算很好玩的。

注意：遊戲只有一次機會。■

（這裏把光點寫作蘋果）

```
0  LOMEM: 16384:P1 = 193:P2 = 149
   :P3 = 218:P4 = 136: DIM ST%(
   800,1)
1  TEXT : HOME : FLASH : SPEED= 2
   55
2  HTAB 16: PRINT "DEVIL SNAKE": NORMAL
3  VTAB 3: PRINT "1- JOYSTICK": PRINT
   "2- KEYBOARD": PRINT : PRINT
   "WHICH (1/2) ? ";: GET PL$:P
   L =  VAL (PL$): IF PL < 1 OR
   PL > 2 THEN  PRINT : GOTO 3
4  PRINT PL: PRINT : PRINT "LEVEL
   -1 (EASY)"
5  PRINT "LEVEL-2 (HARD)": PRINT
   "LEVEL-3 (HARDER)"
6  VTAB 12: HTAB 1: PRINT "WHICH
   LEVEL (1-3) ? ";: GET LE$:LE
    =  VAL (LE$): IF LE < 1 OR
   LE > 3 THEN 6
7  PRINT LE: IF PL = 1 THEN 10
8  VTAB 14: PRINT "WANT TO DEFINE
    KEYBOARD ? (Y/N) ? ";: GET
   D$: IF D$ = "Y" THEN  GOSUB
   500
10  HOME : HGR
20  HCOLOR= 3: ROT= 0: SCALE= 1
35  RESTORE
40  FOR X = 1 TO 23: READ P: POKE
    767 + X,P: NEXT
50  DATA  2,0,6,0,13,0,45,53,63,5
    5,45,53,0,45,45,53,54,62,63,
    63,36,36,0
60  POKE 232,0: POKE 233,3
65 EA = 0: ON (LE - 1) GOTO 80,90

70  FOR X = 72 TO 204 STEP 4: DRAW
    1 AT X,2: DRAW 1 AT X,155: NEXT

74  FOR Y = 3 TO 154 STEP 3: DRAW
    1 AT 72,Y: DRAW 1 AT 204,Y: NEXT

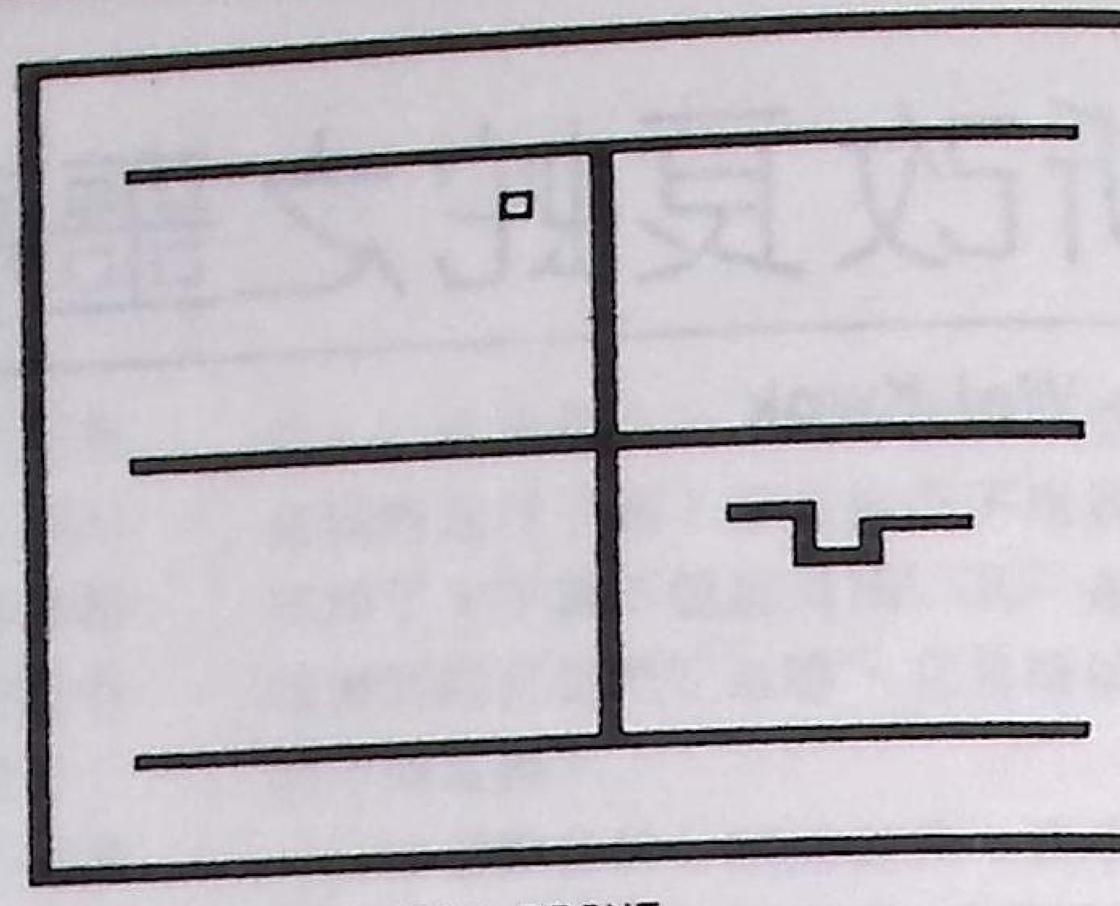
75  FOR X = 92 TO 184 STEP 4: DRAW
    1 AT X,77: NEXT
77  GOTO 110
80  FOR X = 52 TO 224 STEP 4: DRAW
    1 AT X,2: DRAW 1 AT X,155: NEXT

84  FOR Y = 3 TO 154 STEP 3: DRAW
    1 AT 52,Y: DRAW 1 AT 224,Y: NEXT

85  FOR X = 72 TO 204 STEP 4: DRAW
    1 AT X,77: NEXT : FOR Y = 23
     TO 134 STEP 3: DRAW 1 AT 14
    0,Y: NEXT
87  GOTO 110
90  FOR X = 32 TO 244 STEP 4: DRAW
    1 AT X,2: DRAW 1 AT X,155: NEXT

94  FOR Y = 3 TO 154 STEP 3: DRAW
    1 AT 32,Y: DRAW 1 AT 244,Y: NEXT

95  FOR X = 52 TO 224 STEP 4: DRAW
    1 AT X,77: DRAW 1 AT X,23: DRAW
    1 AT X,134: NEXT : FOR Y = 2
    3 TO 134 STEP 3: DRAW 1 AT 1
    40,Y: NEXT
110 XC = 0:Y = 149:L = 1:N = 1:C =
    1:DI = 1: ON (LE - 1) GOTO 1
    14,117
111 X = 80: GOTO 119
114 X = 60: GOTO 119
117 X = 40
119  VTAB 23: HTAB 10: PRINT "SCO
    RE: ";LL: VTAB 21: HTAB 10: PRINT
    "APPLES LEFT: 10"
120 ST%(N,0) = X:ST%(N,1) = Y
130 N = N + 1: IF N = 801 THEN N =
    1
140  XDRAW 1 AT X,Y:CL =  PEEK (2
    34): XDRAW 1 AT X,Y
150  IF CL = 0 THEN 440
160  IF CL <  > 12 THEN  PRINT  CHR$
    (7);:XC = 999:LL = LL + (5 *
    LE):C = C + 5:EA = EA + 1: VTAB
    23: HTAB 10: PRINT "SCORE: "
    ;LL;: IF EA > 9 THEN LL = LL
     + 30: GOTO 600
165  VTAB 21: HTAB 10: PRINT "APP
    LES LEFT: ";10 - EA;" "
170  IF EA < 11 THEN  GOSUB 330
175 SU =  PEEK ( - 16336)
180  XDRAW 1 AT X,Y
190  IF C = 0 THEN  XDRAW 1 AT ST
    %(L,0),ST%(L,1):L = L + 1: IF
    L = 801 THEN L = 1
```



```
200  IF C > 0 THEN C = C - 1
205  IF PL = 1 THEN 272
210 K =  PEEK ( - 16384)
220  IF K < 128 THEN 280
230  POKE  - 16368,0
240  IF K = P1 THEN DI = 1
250  IF K = P2 THEN DI = 2
260  IF K = P3 THEN DI = 3
270  IF K = P4 THEN DI = 4
271  GOTO 280
272  IF  PDL (1) < 20 THEN DI = 1
273  IF  PDL (0) > 235 THEN DI =
     2
274  IF  PDL (1) > 235 THEN DI =
     3
275  IF  PDL (0) < 20 THEN DI = 4
280  IF DI = 1 THEN Y = Y - 3
290  IF DI = 2 THEN X = X + 4
300  IF DI = 3 THEN Y = Y + 3
310  IF DI = 4 THEN X = X - 4
311  IF Y <  > 2 OR EA <  > 10 THEN
     320
315  XDRAW 1 AT X,Y:CK =  PEEK (2
     34)
317  IF CK = 12 THEN 630
318  GOTO 440
320  GOTO 120
330  IF XC > 1 AND XC <  > 999 THEN
     XC = XC - 1: GOTO 430
335  IF XC = 999 THEN  XDRAW 2 AT
     XX,YY:XC = 0
340  IF XC = 1 THEN 442
343  IF EA = 10 THEN  RETURN
345  HCOLOR= 3
365  ON (LE - 1) GOTO 367,368
366 XO = 125:XT = 76: GOTO 370
367 XO = 165:XT = 56: GOTO 370
368 XO = 205:XT = 36
370 XX =  INT ( RND (1) * XO + XT
     )
380 YY =  INT ( RND (1) * 151 + 3
     )
390  XDRAW 2 AT XX,YY:CL =  PEEK
     (234): XDRAW 2 AT XX,YY
400  IF CL <  > 18 THEN 370
410  DRAW 2 AT XX,YY
420 XC = 50 + (LE * 25)
430  VTAB 22: HTAB 10: PRINT "TIM
     E: ";XC;" ": RETURN
440  VTAB 23: HTAB 1: PRINT "CRAS
     H";: FOR I = 1 TO 5: PRINT  CHR$
     (7);: NEXT
441  GOTO 443
442  VTAB 23: HTAB 1: PRINT "SORR
     Y, TIME OUT";: CALL  - 868
443  FOR I = 1 TO 40: FOR C = 0 TO
     7: HCOLOR= C: DRAW 1 AT X,Y:
     SU =  PEEK ( - 16336): NEXT
     C,I
445  VTAB 24: HTAB 1: PRINT "PLAY
      AGAIN ? (Y/N) ";
450  GET YN$: IF YN$ = "Y" THEN 1
460  IF YN$ = "N" THEN  TEXT : HOME
     : END
470  GOTO 450
500  HOME :Q$ =  CHR$ (124)
510  HTAB 21: PRINT  CHR$ (P1 - 1
     28): PRINT : HTAB 21: PRINT
     Q$: HTAB 21: PRINT Q$: PRINT
     : HTAB 21: PRINT Q$: HTAB 21
     : PRINT Q$: PRINT : HTAB 21:
      PRINT  CHR$ (P3 - 128)
520  VTAB 5: HTAB 19: PRINT "--+-
     -";
521  IF P2 <  > 149 AND P4 <  > 1
     36 THEN 530
522  IF P4 = 136 THEN  HTAB 16: PRINT
     "<-";
523  IF P2 = 149 THEN  HTAB 25: PRINT
     "->"
525  GOTO 540
530  HTAB 17: PRINT  CHR$ (P4 - 1
     28);: HTAB 25: PRINT  CHR$ (
     P2 - 128)
540  VTAB 15: PRINT "IS THAT CORR
     ECT ? (Y/N) ";: GET G$: IF G
     $ <  > "N" THEN  RETURN
541  PRINT "N"
542  DATA   1044,1084,1551,1552,1
     560,1561
543  RESTORE : FOR I = 1 TO 23: READ
     P: NEXT
545  FOR I = 1 TO 6: READ J: POKE
     J,32: NEXT
547 CL = 0
550 CL = CL + 1: IF CL = 5 THEN 5
     00
551  ON CL GOSUB 553,554,555,556
552  GOTO 560
553 V = 1:H = 21: RETURN
554 V = 5:H = 25: RETURN
555 V = 9:H = 21: RETURN
556 V = 5:H = 17: RETURN
560  VTAB V: HTAB H: GET A$
561  IF CL = 2 AND  ASC (A$) = 21
      THEN  PRINT "->":P2 = 149
562  IF CL = 4 AND  ASC (A$) = 8 THEN
      PRINT  CHR$ (8);"<-":P4 = 1
     36
563  IF (CL = 2 AND A$ =  CHR$ (2
     1)) OR (CL = 4 AND A$ =  CHR$
     (8)) THEN 570
564  ON (CL - 1) GOTO 566,567,568
565 P1 =  ASC (A$) + 128: GOTO 56
     9
566 P2 =  ASC (A$) + 128: GOTO 56
     9
567 P3 =  ASC (A$) + 128: GOTO 56
     9
568 P4 =  ASC (A$) + 128: GOTO 56
     9
569  PRINT  CHR$ (8);" ";A$;" "
570  GOTO 550
600  XDRAW 1 AT 140,2
620  GOTO 165
630 LE = LE + 1: IF LE > 3 THEN L
     E = 1
635 XC = 0:EA = 0
640  GOTO 10
650  REM  **************
660  REM  *            *
670  REM  * NEW DEVIL  *
680  REM  *   SNAKE    *
690  REM  *    BY      *
700  REM  *  W.K.LEE   *
710  REM  *            *
720  REM  **************
```

工兵掘地雷

工兵掘地雷這程式，是從一矩陣式的遊戲改良出來，加入了音响、升級制等。顧名思義，玩這遊戲時，你可想像自己是一個工兵，要負責將一份重要的文件帶到司令部去，你開始的位置在左上角，司令部則在右下角「＊」處，在途中你要經過一個地雷陣，如何能順利到達右下角則要運用你的智力、判斷力解除地雷的威脅。

此程式的特點，是能盡量地運用一些基本ＢＡＳＩＣ的簡單指令，減少用那些深奥的技巧，對一些初學電腦的人士較易看得懂，程式中之音响效果，都是利用ＢＡＳＩＣ原有之指令，產生變化；放棄了用機械碼SHAPE TAB—LE等之方法，力求簡潔，程式開始由ＧＯＳＵＢ跳至後段用作地雷陣地構圖，程式玩法說明由900至950行顯示，工兵之移動由Ｉ、Ｊ、Ｋ、Ｍ鍵制控制。

有關基本遊戲之方法，程式在遊戲中都有指示，唯一要提及的是當你愈接近地雷時，螢幕右方是有指示的，當十分接近地雷時，會有一聲警號，這時你可運用Ｆ來炸毁地雷，但如你炸不中，ＢＯＮＵＳ會扣一百分。炸中會奬一百分。不過大家在最初的ＬＥＶＥＬ時，最好運用技巧繞過地雷，保留Ｆ炸地雷之權利，因到升級後地雷多了，你便發覺保留炸地雷之重要。另在前進時，要留意左下角顯示之ＳＴＥＰ的步數，因ＳＴＥＰ步數出現零時，亦同樣表示了GAME OVER。

遊戲在程式磁碟的FILENAME是DEATH MINE。■

```
13   REM  *********************
14   REM  **                 **
15   REM  **   DEATH MINE    **
20   REM  **       BY        **
30   REM  **  HO KAM KEUNG   **
35   REM  **                 **
40   REM  *********************
150   GOTO 960
160   HOME :BN = 0:X = 0:Y = 0:T =
      0: GOSUB 410
170   VTAB Y * 2 + 1: HTAB X * 2 +
      1: PRINT "X"
180   HOME : IF X = 14 AND Y = 7 THEN
      540
190  H = H - 1: IF H = 0 THEN 750
200   PRINT "STEP=[ ";H;"]": PRINT
      TAB( 17);"BONUS=";BN
210   PRINT "DIRECTION: ";: GET DI
      $:T = T + 1
220   VTAB Y * 2 + 1: HTAB X * 2 +
      1: PRINT " ";
230   IF DI$ = "K" AND X < 14 THEN
      X = X + 1: GOTO 290
240   IF DI$ = "M" AND Y < 7 THEN
      Y = Y + 1: GOTO 290
250   IF DI$ = "J" AND X > 0 THEN
      X = X - 1: GOTO 290
261   IF DI$ = "I" AND Y > 0 THEN
      Y = Y - 1: GOTO 290
270   IF DI$ = "F" THEN  GOSUB 400
      :BN = BN - 100: GOTO 320
280  T = T - 1: GOTO 170
290   VTAB Y * 2 + 1: HTAB X * 2 +
      1: PRINT "X"
300   IF P(X,Y) THEN P = 0: GOSUB
      460: GOTO 800
310   GOSUB 410: GOTO 180
320   IF F <  = 0 THEN 180
330   HOME : PRINT "COORDINATE - (
      X   ,Y)
340   VTAB 18: HTAB 16: GOSUB 500:
      K = G
350   VTAB 18: HTAB 20: GOSUB 500:
      L = 7 - G: IF G > 7 THEN 350
360   IF X = K AND Y = L THEN 330
370   VTAB L * 2 + 1: HTAB K * 2 +
      1: INVERSE : PRINT "+";: NORMAL
380   IF P(K,L) THEN  PRINT  CHR$
      (7) + CHR$ (7) + CHR$ (7) +
      CHR$ (7):Z = Z + 200:BN = B
      N + 200
390  Z = Z - 100:F = F - 1: VTAB 1
      2: HTAB 39: PRINT "  ": VTAB
      12: HTAB 39: PRINT F:P(K,L) =
      0: GOTO 180
400   VTAB Y * 2 + 1: HTAB X * 2 +
      1: PRINT "X";: RETURN
410  P = 15: FOR K = 0 TO 7: FOR L
      = 0 TO 14
420   IF P(L,K) = 0 THEN 450
430  P1 =  ABS (Y - K) +  ABS (X -
      L)
440   IF P1 < P THEN P = P1
450   NEXT L,K
460   VTAB 16: HTAB 36
470   IF P < 10 THEN  PRINT " ";
480   IF P < 3 THEN  PRINT  CHR$ (
      7);: FLASH
490   PRINT P;: NORMAL : RETURN
500   INPUT G$: IF G$ >  = "0" AND
      G$ <  = "9" THEN G =  VAL (G
      $): GOTO 530
510   IF G$ >  = "A" AND G$ <  = "
      E" THEN G =  ASC (G$) - 55: GOTO
      530
520   GOTO 500
530   PRINT "X=";G$;: RETURN
540   HOME : PRINT  TAB( 9): FLASH
      : PRINT "*** PERFECT ***": PRINT
      : PRINT "": PRINT : NORMAL
550  HO = 99: PRINT "  "
560   PRINT "YOU WIN THE GAME IN "
      ;T;" STEPS."
570   FOR I = 1 TO 1500: NEXT
580   FOR I = 1 TO 10: PRINT  CHR$
      (7) + CHR$ (7) + CHR$ (7) +
      CHR$ (7): NEXT
590  Z = (80 - T) * NB: PRINT "YOU
```

```
     R SCORE=";Z
600  FOR I = 1 TO 2500: NEXT
610  FOR Y = 0 TO 7: FOR X = 0 TO
     14
620  IF P(X,Y) THEN  VTAB Y * 2 +
     1: HTAB X * 2 + 1: PRINT "+"
     ;
630  NEXT X,Y
640  FOR I = 1 TO 3500: NEXT I: GOSUB
     770
650  IF HO = 99 THEN 700
660  TEXT : HOME : PRINT "PLAY AG
     AIN? Y/N"
670  GET U$
680  IF U$ = "Y" THEN 150
685  IF U$ <  > "N" THEN 670
690  END
700 NB = 4 + NB: IF NB > 50 THEN
     NB = 50
710  HOME : PRINT " ": PRINT "GOT
     O NEW LEVEL [ ";NB;" ] BOMBS
     "
720  PRINT "GOOD LUCK!"
730  FOR I = 1 TO 3200: NEXT I:X =
     0:Y = 0:T = 0:HO = 0
740 F = F + 2: VTAB 12: HTAB 39: PRINT
     F: GOTO 1220
750  PRINT "STUPID!!!": FOR I = 1
      TO 190:M =  PEEK ( - 16336)
      +  PEEK ( - 16336): NEXT : PRINT
      CHR$ (7) +  CHR$ (7) +  CHR$
     (7)
755  INVERSE : PRINT "......GAME
     OVER........": NORMAL
760  PRINT "YOUR STEP IS OVER STA
     NDAR!??": FOR I = 1 TO 3500:
      NEXT : GOTO 660
770  FOR Y = 0 TO 7: FOR X = 0 TO
     14
780  IF P(X,Y) THEN  VTAB Y * 2 +
     1: HTAB X * 2 + 1: PRINT ";"
     ;
790 P(X,Y) = 0: NEXT X,Y: RETURN

800  FOR K = 1 TO 10: HOME
805  COLOR= 9
810  PRINT  TAB( 6);" BBBB    OOO
        M     M  BBBB"
820  PRINT  TAB( 6);" B    B  O
     O  MM MM  B    B"
830  PRINT  TAB( 6);" B    B  O
     O  MM MM  B    B"
840  PRINT  TAB( 6);" BBBB   O
     O  M M M  BBBB"
850  PRINT  TAB( 6);" B    B  O
     O  M     M  B    B"
860  PRINT  TAB( 6);" B    B  O
     O  M     M  B    B"
870  PRINT  TAB( 6);" BBBB    OOO
        M     M  BBBB";
880  FOR A = 1 TO 20:M =  PEEK ( -
     16336) -  PEEK ( - 16336): NEXT

890  NEXT : HOME : PRINT : PRINT
      TAB( 16);"GAME OVER": GOTO
     590
900  HOME : PRINT "YOU STAND AT
[
     X] YOUR MISSION IS [*].KEY C
     ONTROL I (UP),M (DOWN),J (LE
     FT),K (RIGHT)"
905  FOR I = 1 TO 3000: NEXT : VTAB
     15: HTAB 29: PRINT "*": HOME

910  PRINT "SCREEN WILL SHOW THE
     NEAREST DISTANCE FROM THE BO
     MBS AND YOU HAVE SOME BULLET
     S TO TEST THE BOMBS.KEY CONT
     ROL IS <F>. "
915  FOR I = 1 TO 300: NEXT
920  PRINT "GOD BLESS YOU!": FOR
     I = 1 TO 5500: NEXT
930  RETURN
940  GR : COLOR= 14
960  TEXT : HOME
970  FOR Y = 0 TO 7: VTAB Y * 2 +
     1
980  FOR X = 0 TO 14: HTAB X * 2 +
     1
990  PRINT ".";: NEXT X: PRINT "
     ";
1000  INVERSE : PRINT (7 - Y); CHR$
     (10); CHR$ (8);" ": NORMAL :
      NEXT Y
1010  INVERSE : FOR K = 0 TO 14:K
     $ =  CHR$ (48 + K)
1020  IF K > 9 THEN K$ =  CHR$ (K
     + 55)
1030  PRINT K$;" ";: NEXT : PRINT
     " ";: NORMAL
1040  POKE 32,32: POKE 33,8: HOME
     : CLEAR : DIM P(14,7):F = 5
1050  PRINT "D"
1060  PRINT "I"
1070  PRINT "R      "
1080  PRINT "E    I"
1090  PRINT "C J-X-K"
1100  PRINT "T    M"
1110  PRINT "I"
1120  PRINT "O"
1130  PRINT "N"
1140  PRINT ":": PRINT
1150  PRINT "FIRE= 5"
1160  PRINT "  BOMB"
1170  PRINT "DISTANCE"
1180  HTAB 2: PRINT "[ ";" ";"   ]
     ";: NORMAL
1190  TEXT : POKE 34,17
1192  VTAB Y * 2 + 1: HTAB X * 2 +
     1: PRINT "X"
1195  HOME : PRINT "INSTRUCTION (
     Y/N)?";: GET I$: IF I$ = "Y"
      THEN  GOSUB 900
1200  HOME : INPUT "HOW MANY BOMB
     S (5-30)? ";H$:NB =  VAL (H$
     ): IF NB < 5 OR NB > 30 THEN
     1200
1210  IF NB > 10 THEN F =  INT (N
     B / 8) + F: VTAB 12: HTAB 39
     : PRINT F
1220 H = 25 + NB
1230  FOR R = 1 TO NB
1240 X =  INT ( RND (1) * 15):Y =
      INT ( RND (1) * 8)
1250  IF P(X,Y) <  > 0 OR X + Y =
     0 OR X + Y = 21 THEN 1240
1260 P(X,Y) = 1: NEXT
1280  GOTO 160
```

圖形建造師

簡化繪製圖形表過程

APPLE Ⅱ與其它電腦優勝的原因是因爲它擁有強效的高解像度繪圖功能，而且由於SHAPE TABLE（圖形表的使用方便，使動畫的創作更加生動有趣，我相信這是APPLE Ⅱ有著這麽多遊戲的主要原因，不過在設計圖形表時十分費時，若果有錯誤，便需要重頭再開始，很是麻煩。本程式便是設計來作這個功能用的。它可以用來創造，觀看，編輯，貯存圖形表，而經由本程式所造出來的圖形表都是以標準的十六進制代碼格式來貯存，以後你可以用在任何的APPLESOFT或機械語言程式之中。

建立圖形表方法

有關如何去建立圖形表，大家可以參考REFERENCE MANUAL 來得到詳細的資料，這裏我將一些要點告訴大家。首先要先繪出圖形，然後選取一個開始點，用箭咀將一點與一點之間連接，但彎角只用90°箭咀來連接，這些就是圖形的向量（VECTOR），而這些箭咀必須轉換爲3個位的二進制數字，然後再轉爲8 BIT BYTE，(即十六進制代碼），這是很麻煩的事，然而，圖形建造師可以替你自動去完成。

當你完成編碼的工作後，你要建立SHAPE TABLE INDEX，這個INDEX 是告訴圖形表的位址。

圖形建造師能夠將你繪畫在螢幕上的圖形自動轉換爲圖形表，它的主功能如下：

第一項功能使你可以建立圖形表，電腦會問你要使用TURTLE或LINE 模式，若果使用LINE 模式時，螢幕會出現HLIN FROM　　PLOT，意指你現在處於水平線模式，現在你按入5及RET，現在會出現TO，按入9〔RET〕，螢幕會再出現ON，按7〔RET〕，這時會繪出一條水平線由5，7到9，7。

若果你按入V〔RET〕螢幕會出現

VLIN FROM　　PLOT，即現在你正處於垂直線模式之中，你可以自己按入數字試試效果。

若果你按E〔RET〕便會出現

HLIN FROM　　ERASE，會將所繪的線清除。

按入F可以進入FREE CURSOR 模式，你可以用I，J，K，M來移動CURSOR，它們經過的地方會留下紅色點，若你按入E來移動CURSOR，可以把它們清除。

當我們完成繪畫之後，可以按Q離開，程式會計算圖形表，程式並會顯示LOCATING SHAPE EDGES，程式會自動找尋圖形的最上方，最下方等等，一會兒後會顯示CODING PLOV VECTORS及POKING SHAPE DEFINITION，螢幕上的圖形會出現一個白點，即表示這個圖形會由這個白點開始繪出。

顯示模式

此時若果你按RETURN，再按入SCALE及ROTATION，並選擇顏色，你便可以看到高解像圖形的出現，此時若再按入RETURN程式會問你要否重繪圖形，若你按入Y，你可以輸入新座標再繪。若按入N的話，電腦便會問你要否修改圖形，按Y你便可以回到修改模式去，若你滿意的話你可以按N。現在電腦會問你是否要保留此圖形作爲另一圖形的BASE，假如你要製作動畫的話便按入"Y"並繼續繪畫，此時你會進入輸入模式之中。

TURTLE 模式

假如你選擇TURTLE模式的話，螢幕會顯示MOVE TO START AND PRESS "P"此時你可以用I，J，K，M鍵去移動CURSOR，若果你按入P，螢幕便會顯示PLOT ON X=xx Y=yy，這是告訴你繪圖模式已開始，CURSOR 經過的地方會當作爲圖形的一部份，按入E鍵可以終止繪畫模式，而使用ESC鍵可以使你將已繪畫的前一點刪除。你可以用它來作爲除錯之用。

自動 POKE

這是本程式的第二項功能，當你使用本程式來建立圖形表時，圖形表會貯存在高解像度的第一頁內，當你將這些圖形表載入到自己的程式上使用時，你必須載入到其它位置去，而它們的位址必須POKE入232及233上。

而本功能是設計來給你做這項工作時更加方便，程式會問你需要將圖形表放到那裏，你回答後程式會顯示出你程式中需要用何種POKE的形式在你的程式之中。

抱歉由於列表太長，這裏只刊登本程式玩法。

本程式在程式磁碟的檔案名字是SHAPE BUILDER。

LOCK IT UP 破解方法

LOCK—IT—UP是一個很利害的 PROTECTION PROGRAM。相信很多的讀者都嘗試過它的利害，但現在不怕了，本人將提供各位讀者一個破解方法，首先用DISK MUNCHER 1.0或 2.0 將被LOCK—IT—UP鎖了的磁碟復製一次，然後BOOT起BACK—IT UP〕〔十2.4將0C這項參數改爲40，然後復製第0條磁軌，記着只是復製爲0這一條磁軌。跟着先貼上保護標貼（WRITE—PROTECT），然後才可 BOOT 機。依照上述方法，如果不能成功的話，可能是磁碟機轉速上出了問題或者是這片磁碟根本不是LOCK—IT—UP 保護出來的，大家不妨再重覆上述的步驟。■

有多項功能的
AMPERSAND工具程式

EDWARD LEE

AMPERSAND這個程式，是一個非常實用而有趣的程式。它是利用APPLESOFT的& 功能來執行的。（詳閱THOMAS的專欄）由於要便利其中幾個主要命令的執行，我把程式放置到$4000至$45A4，所以各位如果要用到高解像圖第二頁，則用完後再BRUN程式便可。

現在讓我先講一講程式的操作方法。BRUN後，每當APPLESOFT遇上&鍵，便會跳至$ 4000(即程式的開頭地址）立刻清除TEXT畫面（$ 4019— JSR $FC58）各位如不想打入&後HOME，可以把$ 4019至$401B 三個地址全部改爲EA（NOP）便行。然後電腦如再發現有命令碼，便另作執行。但其實各位在&之後打入任何命令碼以外的字串，APPLESOFT也不會賭你一句：SYNTAX ERROR，因爲我用了一個簡單的 6502 指令：JMP $ 03D0，即跳至APPLESOFT的RE—ENTRY POINT。（這僅須三BYTES而不必用到DATA副程式）

AMPERSAND有八個功能，分別爲：C）ATALOG，E）DIT，H）I—RES，H2）HI—RES PAGE 2，M）ONTIOR，R）EM—STATEMENT；V）OLUME—HEADING 及？）COMMAND ENQUIRY。C，E，H，H2等等就是各指令的命令碼。

最初的幾個命令，十分簡單：

①& C即CATALOG

②首次& E等如POKE33，33（方便EDIT程式）再& E就是POKE33，40還原闊度；再& E就等如POKE33，33——這樣梅花間竹地工作。

③& H和& H2分別顯示高解像圖一和二頁而不清洗畫像。

④& M等如CALL—151 進入APPLE的MONITOR PROGRAM。

但是& R就異常獨特了。BEAGLE BROTHERS的實用BASIC程式相信各位都有用過或略有所聞了，但各位又曾否LIST過他們的程式呢？如果有，那麼各位一定會發覺到在那些LISTINGS 內，有一些頗爲奇妙的現象。如在他們的BYTE ZAP 程式裏：（見圖1）留意在行90之前就有一段類似REM的字串，但是它美觀得多，因爲旣沒有行數，又沒有REM指令！各位無論明不明白個中奧妙，現在也可以輕而易舉地製造出這種效果——利用REM—STATEMENT 指令！

譬如說你希望在你的程式中行100 前加一句INITIALIZE，那麼& R後，AMPERSAND 便會問你這行數，這時你應輸入"10"。跟着的問題是請你把字串打入（最多三十三字），照做可也。這時AMPERSAND 已完成了它一半的工作，你只要把跟着"]"的那一句子用向右鍵"→"輸入，之後CALL 17345（這也已經爲你印在螢幕上了）你就會看見反白的DONE .字樣，表示電腦已完成工作。現在LIST1，100(圖2）效果一目了然。

圖（1）

```
]
]RUN

]LIST 1,90

==============
BYTE ZAP
BY BERT KERSEY

COPYRIGHT 1982
BEAGLE BROS
==============

90 SS = 1: REM PRINTER.SLOT
```

圖（2）

```
*
]RUN FIG.2

]LIST 1,100

INITIALIZE

100  TEXT : HOME :S =  - 16336:R$
     = " [RETURN]":O$ =  CHR$ (1
    62):M$ = "]":S$ = " ":N = 49
    385:F = 49384:X$ = "..."
```

第六個命令也是很有趣的。各位每次 CATALOG磁碟都會看見DISK VOLUME 254這句，相信大家都知道改VOLUME NO.只會帶來不便，可但"DISK VOL—UME"這字串是可以隨便更改的。辦法有很多，其中一個就是用 AMPERSAND的VOLUME HEADING 功能。打入& V後，放入磁碟，按一鍵，螢幕上便會顯示當時那張磁碟的VOLUME HEADING是什麽。然後只要打入你喜歡的句語，再按一鍵，工作便完成了。不過有一件事要注意的，就是若你這時& C還看見舊的VOLUME HEADING，是正常的，因爲DOS只會記着最近一次BOOT碟的那個字串。所以如果你想看一看是否"變身"成功，可以& V或者BOOT碟。

而每當各位忘記了AMPERSAND 有什麽功能時，&？就是了。

有一點關於& C筆者希望補充一下：CATALOG磁碟沒有用到它在DOS的進入點$A56E，原因是JSR $ A56E所做出來的效果十分慢。各位不信可& M後打入A56

ＥＧ和打入CATALOG 比較一下，筆者就是用了後者。

連同 AMPERSAND 有一個獨立的BASIC 程式A—MPER—DEMO ，作爲示範的工作。這程式生動而容易明白。其中常數N和F更蠱惑得很；而最特別的一點，就是在行 1070 尾的Ｘ！它是百份百正確的，作用就留待各位想一想了。（讀者要參看ＤＥＭＯ可從即期程式磁碟中找到）

最後筆者僅藉此機會多謝宗林精神上的支持，並希望各位在ＨＥＬＬＯ中加多一句BRUM AMPERSAND，多多運用！

本文主程式AMPERSAND本身只有幾個ＳＥＣＴＯＲ長。但筆者編寫有一個ＤＥＭＯ程式來詳細講述本程式的用途示範。不過由於ＤＥＭＯ程式長度超過有35個ＳＥＣＴＯＲ，爲節省篇幅，故此不擬在此將ＤＥＭＯ程式 LISTING 列出，不過，讀者可在即期的程式磁碟中找到。

本文程式包括有7 個 ＦＩＬＥ ，它們在程式磁碟中社 FILENAME 是：

```
AMPERSAND
AMPER-DEMO
LEVEL.PIC
FIG.1
BYTE ZAP
FIG.2
INITIALIZE
```

```
]
*4000.45A6

4000- A9 4C 8D F5 03 A9 19 8D
4008- F6 03 A9 40 8D F7 03 A9
4010- 28 85 21 20 58 FC 4C 37
4018- 45 20 58 FC AD 01 02 C9
4020- 43 D0 1A A2 00 BD 33 40
4028- 20 ED FD E8 E0 0A D0 F5
4030- 4C D0 03 8D 84 C3 C1 D4
4038- C1 CC CF C7 8D C9 45 D0
4040- 1E AD 5E 40 D0 0C A9 21
4048- 85 21 A9 01 8D 5E 40 4C
4050- D0 03 A9 28 85 21 A9 00
4058- 8D 5E 40 4C D0 03 00 C9
4060- 48 D0 1C AD 02 02 C9 32
4068- F0 0F AD 54 C0 AD 52 C0
4070- AD 57 C0 AD 50 C0 4C D0
4078- 03 AD 55 C0 4C 6D 40 C9
4080- 4D D0 03 4C 69 FF C9 52
4088- D0 42 A2 00 B5 67 9D A3
4090- 40 E8 E0 08 D0 F6 A5 AF
4098- 8D AB 40 A5 B0 8D AC 40
40A0- 4C AD 40 00 00 00 00 00
40A8- 00 00 00 00 00 A9 40 85
40B0- 68 A9 D0 85 67 A9 43 85
40B8- B0 85 6A 85 6C 85 6E A9
40C0- 6D 85 AF 85 69 85 6B 85
40C8- 6D 4C 66 D5 4C 0A 44 00
40D0- DA 40 0A 00 A5 AB 31 31
40D8- 30 00 50 41 14 00 BA 22
40E0- 41 54 54 45 4E 54 49 4F
40E8- 4E 3A 20 57 48 49 4C 45
40F0- 20 49 4E 20 54 48 49 53
40F8- 20 4D 4F 44 45 2C 20 44
4100- 4F 20 4E 4F 54 20 20 20
4108- 45 4E 54 45 52 20 46 41
4110- 4C 53 45 20 44 41 54 41
4118- 53 20 41 4E 44 20 22 3B
4120- 3A 9E 3A BA 22 44 4F 20
4128- 4E 4F 54 20 50 52 45 53
4130- 53 20 52 45 53 45 54 22
4138- 3B 3A 9D 3A BA 22 54 4F
4140- 20 51 55 49 54 20 49 4E
4148- 50 55 54 20 27 51 27 00
4150- 9F 41 1E 00 BA 3A 84 22
4158- 45 4E 54 45 52 20 54 48
4160- 45 20 4C 49 4E 45 20 4E
4168- 4F 2E 20 4A 55 53 54 20
4170- 41 46 54 45 52 20 54 48
4178- 45 20 4F 42 4A 45 43 54
4180- 52 45 4D 2D 53 54 41 54
4188- 45 4D 45 4E 54 3A 22 3B
4190- 4C 24 3A AD 4C 24 D0 22
4198- 51 22 C4 31 30 30 00 E9
41A0- 41 28 00 4C D0 E5 28 4C
41A8- 24 29 C9 31 3A 4D D0 32
41B0- CA 28 E3 28 E4 28 4C 29
41B8- 29 C8 36 29 3A 4D 53 D0
41C0- D3 28 4C CB 32 35 36 29
41C8- 3A 4C 53 D0 4C C9 32 35
41D0- 36 CA 4D 53 3A B9 31 37
41D8- 33 34 36 2C 4C 53 3A B9
41E0- 31 37 33 35 30 2C 4D 53
41E8- 00 59 42 32 00 BA 3A 84
41F0- 22 45 4E 54 45 52 20 54
41F8- 48 45 20 4F 42 4A 45 43
4200- 54 20 52 45 4D 2D 53 54
4208- 41 54 45 4D 45 4E 54 20
4210- 28 4D 41 58 2E 20 33 33
4218- 20 43 48 41 52 41 43 54
4220- 45 52 53 29 3A 22 3B 53
4228- 24 3A 53 24 D0 E8 28 53
4230- 24 2C 33 33 29 3A 53 D0
4238- E3 28 53 24 29 3A B9 33
4240- 34 2C 31 30 3A 97 3A B9
4248- 33 34 2C 30 3A AD 53 24
4250- D0 22 51 22 C4 31 30 30
4258- 00 D8 42 3C 00 A2 31 32
4260- 3A BA 22 57 49 54 48 20
```

```
4268- 54 48 45 20 52 49 47 48
4270- 54 20 41 52 52 4F 57 20
4278- 4B 45 59 20 27 2D 3E 27
4280- 20 4D 4F 56 45 20 54 48
4288- 45 20 20 43 55 52 53 4F
4290- 52 20 4F 56 45 52 20 54
4298- 48 45 20 46 4F 4C 4C 4F
42A0- 57 49 4E 47 20 4C 49 4E
42A8- 45 20 41 4E 44 20 50 52
42B0- 45 53 53 52 45 54 55 52
42B8- 4E 2C 20 54 48 45 4E 20
42C0- 43 41 4C 4C 20 31 37 33
42C8- 34 35 20 54 4F 20 43 4F
42D0- 4E 54 49 4E 55 45 2E 00
42D8- FD 42 46 00 B9 31 37 33
42E0- 38 31 2C 4D CB 32 3A B9
42E8- 31 37 33 38 37 2C 4D C9
42F0- 31 3A B9 31 37 33 39 31
42F8- 2C 4D C8 53 00 25 43 50
4300- 00 BA 3A BA 22 20 22 4C
4308- 22 20 52 45 4D 22 3B 3A
4310- 81 49 D0 31 C1 4D 3A BA
4318- 22 20 22 3B 3A 82 3A BA
4320- 53 24 22 26 00 39 43 5A
4328- 00 BA 3A BA 22 5D 43 41
4330- 4C 4C 20 31 37 33 34 35
4338- 00 4A 43 64 00 8C 31 37
4340- 33 32 32 3A A2 31 35 3A
4348- 80 00 6B 43 6E 00 BA 3A
4350- BA 22 45 52 52 4F 52 20
4358- 45 4E 43 4F 55 4E 54 45
4360- 52 45 44 2E 22 3A AB 31
4368- 30 30 00 00 00 00 00 00
4370- 00 00 00 00 00 00 00 00
4378- 00 00 00 00 00 00 00 00
4380- 00 00 00 00 00 00 00 00
4388- 00 00 00 00 00 00 00 00
4390- 00 00 00 00 00 00 00 00
4398- 00 00 00 00 00 00 00 00
43A0- 00 00 00 00 00 00 00 00
43A8- 00 00 A2 00 BD A3 40 95
43B0- 67 E8 E0 08 D0 F6 AD AB
43B8- 40 85 AF AD AC 40 85 B0
43C0- 60 A9 00 85 50 A9 00 85
43C8- 51 20 1A D6 A5 9C 85 01
43D0- A5 9B 18 69 05 85 00 A5
43D8- 01 69 00 85 01 A9 08 A0
43E0- 00 91 00 C8 C0 00 D0 F9
43E8- A9 0D A0 00 91 00 A0 00
43F0- 91 00 A0 00 B9 02 44 20
43F8- ED FD C8 C0 07 D0 F5 4C
4400- D0 03 8D 04 0F 0E 05 2E
4408- 8D 00 C9 56 D0 75 A2 18
4410- BD 1B 44 20 ED FD CA D0
4418- F7 F0 19 00 BA A9 D4 C9
4420- D5 D1 A0 CF D4 A0 D1 A8
4428- A0 CB D3 C9 C4 A0 D4 D2
4430- C5 D3 CE C9 A9 BA 85 33
4438- 20 35 FD C9 D1 D0 06 20
4440- 8E FD 4C D0 03 A9 44 A0
4448- 50 20 D9 03 4C 62 44 00
4450- 01 60 01 00 02 02 FB B7
4458- 00 50 00 00 01 00 00 60
4460- 01 00 A2 13 BD 6E 44 20
4468- ED FD CA D0 F7 F0 17 A0
4470- BA C7 CE C9 C4 C1 C5 C8
4478- A0 D4 CE C5 D2 D2 D5 C3
4480- 8D 8D 00 4C 33 45 A2 0C
4488- BD AE 50 20 ED FD CA D0
4490- F7 A9 A0 A2 0C 9D AE 50
4498- CA D0 FA A2 18 BD A7 44
44A0- 20 ED FD CA D0 F7 F0 19
44A8- BF A9 D4 C9 D5 D1 A0 CF
44B0- D4 A0 11 A8 A0 CF D4 A0
44B8- C5 C7 CE C1 C8 C3 8D 8D
44C0- 00 A9 A0 85 33 20 6A FD
44C8- AD 00 02 C9 91 D0 03 4C
44D0- D0 03 A0 0B A2 00 BD 00
44D8- 02 C9 8D F0 09 99 AF 50
44E0- 88 E8 E0 0C D0 F0 20 8E
44E8- FD A2 18 BD 1B 44 20 ED
44F0- FD CA D0 F7 A9 BA 85 33
44F8- 20 35 FD C9 D1 D0 06 20
4500- 8E FD 4C D0 03 A9 02 8D
4508- 5C 44 A9 44 A0 50 20 D9
4510- 03 A9 01 8D 5C 44 A2 0E
4518- BD 23 45 20 ED FD CA D0
4520- F7 4C D0 03 8D 2E 02 0F
4528- 0A 20 06 0F 20 04 0E 05
4530- 8D 8D 00 C9 BA D0 EA A2
4538- 00 BD 47 45 D0 02 F0 58
4540- 20 ED FD E8 4C 39 45 C5
4548- C4 D7 C1 D2 C4 A0 CC C5
4550- C5 A7 D3 A0 C1 CD D0 C5
4558- D2 D3 C1 CE C4 A0 C9 D3
4560- A0 CE CF D7 A0 C9 CE D3
4568- D4 C1 CC CC C5 C4 8D C1
4570- CE C4 A0 D2 C5 C1 C4 D9
4578- A0 D4 CF A0 C6 D5 CE C3
4580- D4 C9 CF CE BA A0 A6 A0
4588- C3 AC C5 AC C8 AF B2 AC
4590- CD AC D2 AC D6 AC BF 00
4598- A9 BD A2 28 20 ED FD CA
45A0- D0 FA 4C D0 03 8D DB
```

■

要想磁碟機壽命延長，就必須使用這個清潔程式

磁頭清潔器

在衆多電腦週邊設計之中，磁碟機是最重要的一項。除了主機，最多毛病會發生的亦在它身上。假如我們不好好的保養磁碟機，到發覺因爲磁碟機出毛病而影响讀寫磁碟時，爲時已晚。

磁碟機最常見的毛病所在是機件的經常活動而導致的損耗。然而，更加常見的是磁頭黏滿污漬。

大家都會使用一些磁頭清潔套件來清潔磁頭。在使用清潔套件時若能連同本文所附同的清潔程式一同使用，定必能使你的磁碟機長壽耐用。

假如，你並非粗暴地使用磁碟機，你不會有機件損壞的煩惱。不過，積聚在磁頭上的污垢微塵會導致I／O 錯誤，更嚴重的可能會影响磁碟上的數據。通常，這些微塵污垢有數個來源。其中之一是來自空氣中的塵埃。一張磁碟所暴露出來的磁粉表面面積有限，帶來的塵埃不會太多。至於因爲機蓋打開引進的塵埃大家可以忽略，除非工作環境十分惡劣，有嚴重空氣污染。

塵垢最大的來源是在於磁碟本身。當磁碟放入機後，菊花型膠頭壓下磁環。當磁碟轉動時，磁碟，讀寫磁頭及膠頭之間的摩擦會導致磁碟上的微細磁粉飛掉及黏貼磁頭。就算是高質素的磁碟亦會因長時期的使用而有損耗。高質磁碟和劣質磁碟的分別就在於表面分佈的磁粉是否容易磨蝕。硬磁碟機不讓磁頭直接接觸磁碟表面，所以磁碟不會有磨蝕。

筆者不使用電腦時，經常用布蓋住電腦，同時將磁碟機機蓋打開，因爲如此做，膠頭會接觸磁頭而沾汚磁頭。當然，無論你怎樣做足預防功夫；磁頭仍會受沾汚。目前市面有多種的磁頭清潔套件。常見的一種是有多張清潔磁碟及一樽藥水。這些藥水的作用是溶解磁頭上的汚垢，並讓清潔磁碟抹淨。

這些清潔套件用法相當簡易。一般來說，插入磁碟機後，大家會使用CATALOG 來讓磁碟轉動。此擧絕對沒有問題。不過，大家有否發覺，用CATALOG 指令，祇會使用到一部份的清潔碟（包括有DIRECTORY 的幾條磁軌）。事實上，一套清潔磁碟售價並不便宜，只用到少許的清潔碟實在不化算。爲此，筆者編寫了這個程式，讓磁頭能從外緣，横移至內圓心，換言之，整張清潔碟面積也用到。程式可見於列表。

說起來，這個磁頭清潔程式原理相當簡單。執行時，程式會詢問要清潔磁碟機1或磁碟機2。按0會終止程式。程式使用DOS的RWTS子程式來移動讀寫磁頭横過清潔碟表面。

RWTS是READ OR WRITE A TRACK OR SENCT—OR 的簡寫。RWTS是一組的子程式，DOS 用來讀或寫磁區。許多程式都會使用這個RWTS。

由於，筆者無須要讀或寫任何東西，故筆者使用它來"尋找（SEEK）"磁軌。

要使用RWTS，需設定一個稱爲IOCB （INPUT OU—TPUT CONTROL BLOCK）的數目表。這個表包括子程式所有的特別指令。程式中行句800至810的DATA語句有ICOB 所需要的數據。筆者使用POKE敍述句來將這些數據存入記憶之中。由於我們無需要讀或寫任何東西，故此使用0，並讓磁頭横越到所要的磁軌上。

磁頭清潔程式可令磁頭來往＄0軌與＄22軌之間四次。這約花30秒，相信足夠清潔磁頭。假如你經常使用磁碟機，最好每兩週便清理磁碟機一次，以防止塵垢聚積。

本程式在程式磁碟上的FILENAME是DRIVE HEAD CLEANER。■

→

再次爲磁碟命名

莫信

相信各位看完電腦時代第六期的讀者，看到「爲磁碟命名」一文都極感興趣，因爲它包含了一個小小的程式，可以將DOS保留的字頭「DISK V—OLUME」隨你喜歡的更改，當打入了CATALOG 後能夠顯示自己的名字或其他CHARACTER，是相當有滿足感的，但這個程式却有一個缺點，絕不可以將它用MASTER SYSTEM 的MASTER CREAT來改變它作母磁片，CREAT後，被更改的名字便回復「DISK VOLUME」的字樣，就像是沒改變一樣，筆者初用MASTER CREAT時，也是呆了一呆，爲什麽會這樣呢？幾經辛苦查察MASTER CREAT對碟磁片內部的功用和影响，原來它將母片的「DISK VOLU—ME」字樣換了，而這個TITLE原來是被放在TRACK＄2 SECTOR＄2當中，由位址＄AF至＄BA，但我們改動這些位址時，必須留意它們字母的排列次序，而是顚倒次序的，亦即是說由位址＄AF至＄BA是放着「E—MULOV KSID」的ASCII碼，所以利用COPY PROGRAM 來作改動時，要將你所喜愛的名字顚倒塡入，例如你要將「PETER CHAN V」等的字樣做TITLE，便要順序將「V NAHC RETEP」塡入至位址＄AF至＄BA。■

磁頭清潔器

```
10   REM  ***********************
11   REM  *    HEAD CLEANER     *
12   REM  *   BY COMPUHOUSE     *
13   REM  *                     *
14   REM  * COPYRIGHT (C) 1984 *
15   REM  *                     *
16   REM  ***********************
100  REM
110  REM
120  REM  USE THIS PROGRAM WITH
130  REM  A DISK DRIVE HEAD
140  REM  CLEANING KIT TO KIIP
150  REM  YOUR ORIVES CLEAN.
160  REM
170  REM  JUST ENTER "RUN" AND
180  REM  FOLLOW THE DIRECTIONS.
190  REM
200  REM
210 B$ =  CHR$ (7) +  CHR$ (7) +
     CHR$ (7)
220  GOSUB 820
230  REM
240  REM  GET THE DRIVE NUMBER
250  REM
260  TEXT : HOME : POKE  - 16368,
     0
270  VTAB 3: HTAB 12: INVERSE : PRINT
     " HEAD CLEANER ": NORMAL
280  PRINT : PRINT  TAB( 13)"BY C
     OMPUHOUSE"
290  PRINT  CHR$ (7)
300  PRINT : PRINT "  ENTER THE D
     ISK DRIVE NUMBER THAT"
310  PRINT "YOU WANT TO CLEAN (1/
     2,0=END)    ?"
320  PRINT "===> ";: GET DR$
330 DR =  VAL (DR$)
340  IF DR$ = "0" THEN  TEXT : HOME
     : END
350  IF DR <  > 1 AND DR <  > 2 THEN
     PRINT : PRINT : FLASH : PRINT
     "INCORRECT DISK DRIVE";B$: NORMAL
     : FOR I = 1 TO 1000: NEXT I:
     GOTO 260
360  PRINT DR$: PRINT
370  REM
380  REM  PRINT DIRECTIONS
390  REM
400  PRINT B$: PRINT " PUT SOME C
     LEANING FLUID ON A 5 1/4 "
410  PRINT "INCH CLEANING DISKETT
     E AND PLACE IT"
420  PRINT "IN DRIVE NUMBER ";DR$
     ;".  PRESS  ";: INVERSE : PRINT
     " RETURN ": NORMAL
430  PRINT "WHEN YOU ARE FINSHED.
     "
440  REM
450  POKE  - 16368,0
460  IF  PEEK ( - 16384) <  > 13 AND
     PEEK ( - 16384) <  > 141 THEN
     GOTO 460
470  POKE  - 16368,0
480  REM
490  POKE 818,DR
500  POKE 34,10: HOME
510  PRINT : PRINT  CHR$ (7)" RUN
     NING THE HEAD BACK AND FORTH
     "
520  PRINT " TIMES.PRESS ";: INVERSE
     : PRINT "ESC";: NORMAL : PRINT
     "TO STOP"
530  PRINT
540  REM
550  REM  USE THE RTWS "SEEK"
560  REM  M AND TO  RUN THE
570  REM  HEAD ACROSS THE DISK
580  REM
590  FOR CNT = 1 TO 4
600  VTAB 15: PRINT " COUNT =  =
     > ";CNT
610  FOR TRACK = 0 TO 34
620  GOSUB 710
630  NEXT TRACK
640  FOR TRACK = 34 TO 1 STEP  -
     1
650  GOSUB 710
660  NEXT TRACK
670  NEXT CNT
680  PRINT : PRINT "ALL DONE !"
690  FOR I = 1 TO 1500: NEXT I: GOTO
     260
700  REM
710  REM  HEAD SEEK SUBROUTINE
720  REM
730  IF  PEEK ( - 16384) = 27 OR
     PEEK ( - 16384) = 155 THEN
     POP : GOTO 260
740  VTAB 17
750  PRINT "SEEEKING TRACK ==> ";
     TRACK;" "
760  POKE 820,TRACK
770  CALL 837: REM  CALL RWTS
780  RETURN
790  REM  1OCB FOR RWTS
800  DATA  1,96,1,0,0,0,65,3,0,12
     8,0,0,0,0,0,96
810  DATA  1,0,0,239,219,160,48,1
     69,3,32,217,3,96
820  RESTORE
830  FOR I = 816 TO 844
840  READ D: POKE I,D
850  NEXT I
860  RETURN
```

程式拯救者

當我們想清除主機記憶體中的APPLESOFT 程式時，我們都會用到ＮＥＷ這個指令。不過，很多時，我們會因一時手快將主記憶中的程式洗去而感到懊悔，希望把它取回。這裏提供一個非常簡短的程式給你們，只要ＢＲＵＮ此程式，便可將失去的程式取回。有關程式可見列表。

其實本程式所應用的原理相當簡單。當我們打入ＮＥＷ這個指令時，似乎已將程式洗去，其實不是，我們只是將一此POINTER（指標）改變。要取回一個程式，只消將這些指標改變過來便可。以下再詳細舉例解釋其原理。

開閉主機，重新再開機。打入下列句子：

```
10  PRINT "ABCDEFG"
20  GOTO 10
```

打入CALL—151 進入監督程式。

打入800.819, 按〔RET〕

此時螢幕會出現下列的一組數字：

```
0800- 00 10 08 0A 00 BA 22 41
0808- 42 43 44 45 46 47 22 00
0810- 18 08 14 00 AB 31 30 00
0818- 00 00
```

這些數字便是先前打入的二句程式的TOKEN版本。我們稱這些數字爲ＢＹＴＥ。學過機械語言的朋友都應曉得左邊的數目是記憶的位置。數目41至47分別代表Ａ至Ｇ七個字母，而22是代表〝。

大家要留意第二及第三個ＢＹＴＥ，10及08 。這兩個ＢＹＴＥ指示電腦程式的第一句在何位址結束而第二句句子開始。當我們按入ＮＥＷ指令後，這兩個ＢＹＴＥ會消失，而程式似乎是被洗去了。

現在再按69.6E，按〔RET〕

打入AF，B0 按〔RET〕，會有下列數目出現：

```
0069- 1B 08 1B 08 1B 08
00AF- 1A
00B0- 08
```

這是零頁指標用來偵察程式的所在處。當打入ＮＥＷ指令後，它們會重置。有關這個零頁指標，各位可參考APPL—ESOFT MANUAL中的第140頁。

現在按CTRL—C返回BASIC。打入NEW指令將先前輸入的兩句洗去。按入LIST，螢幕不會再有該兩句出現。

CALL—151 進入監督程式。

打入800.819，螢幕會出現下列東西：

```
0800- 00 00 00 0A 00 BA 22 41
0808- 42 43 44 45 46 47 22 00
0810- 18 08 14 00 AB 31 30 00
0818- 00 00
```

在位於800一句上，第二及第三個ＢＹＴＥ現變了00、00。其他保持不變。程式依然是在記憶體中，只是指標改變了。

按69.6E〔RET〕

及AF.B0 〔RET〕

```
0069- 04 08 04 08 04 08 00
00AF- 04
00B0- 08
```

先前所出現的數目會有所改變，1B 08 會改爲04 08。要將洗去的程式取回，只消改變這些指標變回原來的值便可。打入下列幾句：

```
*801:10 08
*AF:1B 08
*69:1B 08 1B 08 1B 08
```

按CTRL—C 走出監督程式，打入LIST，先前失去的程式便可重現眼前。

這個程式執行工作上來速度會快至令人不相信，讀者們不妨試試！

本程式在程式磁碟的檔案是PROGRAM SAVER 。

```
*300.352     ***PROGRAM SAVER**
0300- A2 04 E8 BD 00 08 D0 FA
0308- E8 A9 08 8D 02 08 8E 01
0310- 08 A2 00 A5 00 48 A5 01
0318- 48 A9 08 85 01 A9 01 85
0320- 00 A0 00 E6 00 A5 00 C9
0328- FF D0 02 E6 01 A1 00 D0
0330- F0 C8 C0 03 D0 ED E6 00
0338- A5 01 85 B0 85 6A 85 6C
0340- 85 6E A5 00 85 AF 85 69
0348- 85 6B 85 6D 68 85 01 68
0350- 85 00 60
```

■

「特惠優待讀者證」

再度接受申請

本刊數月前刊登接受申請「特惠優待讀者證」後反應非常熱烈，前後接獲接近萬封的申請書。由於部份申請者未能遵照指示附同一個回郵信封寄來，基於處理技術問題，本刊只好押後處理該批申請書。

雖然，我們經已停止接受申請「特惠優待讀者證」，但仍斷續收到不少遲來的申請書及讀者查詢電話，爲此，本刊決定由即日開始，再度接受「特惠優待讀者證」的申請，申請辦法跟過去一樣，只要塡妥下列表格，連同一個寫上申請人姓名、地址的回郵信封，寄來九龍中央郵政信箱71193 號　電腦時代雜誌社收。信封面請寫上〔申請〕字樣。申請「特惠優待讀者證」是完全免費的，而且絕對沒有任何附帶條件。

持證人可獲得的好處包括如下：

＊特價購買電腦器材
＊特價購買各類週邊咭
＊特價購買程式磁碟、磁帶及雜誌
＊免費索閱「電腦時代讀者之友通訊期刊」
＊獲取外國最新軟件市場動態消息

最後申請截止日期是一九八四年十一月卅日

「特惠優待讀者證」申請表格

本人欲申請「特惠優待讀者證」一張。本人無須付出任何費用。

姓名：＿＿＿＿＿＿

年齡：＿＿＿＿＿＿

職業：＿＿＿＿＿＿

電話：＿＿＿＿＿＿

地址：＿＿＿＿＿＿

申請表格可寄往九龍中央郵政信箱
71193號電腦時代出版社收
信封面請註明「申請」

此份申請表格請用英文塡寫，方便電腦處理。

本人擁有的電腦設備：（請在方格內塡✓）

1 6502主機　：一部 ☐　兩部 ☐　三部或以上 ☐
2 5¼吋磁碟機：一部 ☐　兩部 ☐　三部或以上 ☐
3 印字機　：有 ☐　有 ☐
4 週邊　：80 字行咭 ☐　正考慮購買 ☐
Z－80 咭 ☐　正考慮購買 ☐
ROM咭 ☐　正考慮購買 ☐
說話咭 ☐　正考慮購買 ☐
16 K RAM 咭 ☐　正考慮購買 ☐
其它咭 1 ＿＿＿＿　1 ＿＿＿＿
2 ＿＿＿＿　2 ＿＿＿＿
3 ＿＿＿＿　3 ＿＿＿＿

5 本人由第＿＿＿＿期開始閱讀貴刊。
6 本人每期都購買 ☐ ／間中購買 ☐ 電腦時代程式磁碟。
7 程式磁碟是由：特約經銷處購買 ☐
讀者服務部購買 ☐
從朋友處COPY ☐
8 本人是從：報攤 ☐ 發現電腦時代這份雜誌。
電腦公司 ☐
地鐵電視服務站 ☐
代理書局 ☐
高登／黃金商場 ☐
學校圖書館 ☐
公衆圖書館 ☐
朋友口中 ☐
9 假如閣下家中尙未置有磁碟機，是否希望我們製作出版程式錄音磁帶。是 ☐　不感興趣 ☐
10 本人家中使用的磁碟牌子 1 ＿＿＿＿
（請塡上牌子名稱） 2 ＿＿＿＿
3 ＿＿＿＿

VISICALC的高級應用技術

將VISICALC應用在行政決策上

陳耀南

VISICALC 開創了SPREADSHEET的先河，但在其他情形下的應用彈性似乎受到考驗，例如在行政決策的程式中，與××DECISION，CPA(CRITICAL PATH ANALYZER)等比較時似乎遜色一籌。但筆者以為，這些行政決策軟件的使用說明書又繁又厚，就算免費得來也要花上一段時間來鑽研，極可能到最後才發現不太適宜自己。也許自己動腦筋將問題解決了，還未懂得如何用該等軟件來解決。結果在時間上得不償失。

VISICALC 很容易上手，對於行政決策問題亦可以解決得來（以下簡稱CPA），VISICALC 很適宜用於建立一個網路或有關的項目在月曆上的格式。要建立VISICALC上的CPA系統，首先要訂立報表的格式，一旦完成後一切也可以順序建立——例如審核，改良程式，刪除多餘的部門，建立預算案及需要的勞工數目等，不過在私人應用上CPA也可以使個人的效率提高，以後筆者再為各位介紹，現在以工程上的CPA為例作討論。

規劃工程最先決要訂立一些主要的影响參數，這些參數可以採用方框圖（見圖1）來完成，然後各位可以在VISICALC上的一角輸入作為索引或說明。基本上這只不過是打字問題，不過使用了一些如<<<，>>>，VVV的符號在相關項目上作為分類之用。為了方便觀看我們可以將COLUMN的寬度降低到3（按入/GC3）；而我們首先加入三個CRITICAL項目的符號。

＊ 代表CRITICAL 0 項目（最重要的項目）
一 代表一般項目
F 代表額外時間

第二我們加入主要元素

E 代表開挖 （EXCAVATION）
C 代表注入混凝土（CONCRETING）
B 代表切磚工程（BRICKWORK）
J 代表連結工程（JOINING）

第三我們建立工程所需的步驟：

1 代表第一期工程
2 代表第二期工程
A 代表更改現有工程
E 代表額外工程

現在這些字母可以組合來適應不同的情況，例如

＊EE 代表CRITICAL的額外開挖
一J1 代表一般性第一期連結工程
FB2 代表額外時間完成的 第二期切磚工程。

現在我們試用一張空白的SPREADSHEET來開始，給予名稱後留下數行空白作NOTES，之後要在左方由上至下建立日曆。

☆ 在A11 上按入1
☆ 在A12 上輸入公式（A11＋1）

```
 1   -SC*SAF   -SMF   -IP
         |
 2   -SC*SAF   -SMF   -IP
         |
 3   -SC*SA-SEF   F   F
         |
 4   -SC*SA-SE-SM-SQ-IP
         |
 5   F  *SA-SE-SM-SQ-IP
          \
     V   V   V   V      F
               \
     -->-->-->-->VVV-IP
                 \
 6   -SC-SA-SE-SM*SQ-IP
                  |
 7   F   -SAF   -SM*SQF
                  |
 8   F   -SAF   -SM*SQ-IP
                  |
 9   F   -SAF   -SM*SQF
                  |
10   F   -SAF   F   *SQF
                  |
11   F   F   F   F   *SQF

  COS<==<==<==<==<==<==<==<==<==<==

12

13

14

  GO ==============================

15    -IC-PA-EE

               V---------------CB

16    -IC-PA

17    -IC-PA

18    -IC-PA

19    -IC-PA

20       -PA

21

22

23    XMAXMAXMAXMAXMAXMAXMAXMAXMAXMA

24    XMAXMAXMAXMAXMAXMAXMAXMAXMAXMA
```

圖1

將 VISICALC 應用在行政決策上

☆　將公式以相對格式重複至A 60，假定這些是代表星期，一年有五十二個星期，60個足夠使用了。

☆　假如在一個固定日期開始工程，第二步是加揷聖誕及復活假期或其他勞工假期在內。

假如聖誕是在第二十三個星期，便將"X'MAS"輸入到B 33上並重複到Z 33上。

當CPA建立後你會希望將相關的項目在SPREAD－SHEET上連結，我們可以用→符號或以上的符號來連結它們。

☆　到A 34上按入／IR，這時會空出一行給你，你可以按入"………VVV………>>>"等符號。

☆我們現在可以建立一些代號來分別出不同的部門對工程的參與，例如"DA代表DESIGN BY ARCHITECTS"—SC　代表"SCHEME DESIGN BY ENGINEER"

—SA　代表"SCHEME DESIGN BY ARCHITECTS"

—SE　代表"SCHEME DESIGN BY ELECTRICAL ENGINEER"

—SM　代表"SCHEME DESIGN BY MECHANICAL ENGINEER"

—IP　代表"INITIAL WORK BY PLANNING DEPT."

以上的代號都是以"—"來起始，代表了一般工程，若果是CRITICAL項目時只要將"—"符號改爲"*"便可以了。

```
HOUSE EXTENSION

APPOINT ARCHITECT
      !
      !
PREPARE PLANS-----------------> OBTAIN ESTIMATES
      !                                 !
      !                                 !

SUBMIT FOR PLANNING
    PERMISSION                 APPLY TO BUILDING
      !                         SOCIETY FOR LOAN
      !                                 !
------------------------------------------------
                                   !
                                   !
CLEAR ALL SHRUBS         FINALISE ESTIMATES
  BUILDING AREA            APPOINT BUILDER
      !                        !
      !                        !
CLEAR FURNITURE ------> START BUILDING WORK
AND CARPETS IN
BUILDING AREA
```

圖(二)

圖二顯示出該項工程的成本及圖則需在十一週內完成，而且包括有關的部門在內（以／GC 3格式顯示），我們將"*"符號以黑綫連結起來，代表各部門之間的資料可以在五週內完成，而十一週的時間分配可以清晰表示，對於各部門之間的分工合作可以更協調。各位不妨對圖二多加研究還可以發掘出更多的技巧。■

速成學會VISICALC（下）

編者按：本文是爲有興趣想學用VISICALC軟件的讀者而寫，共分上、下兩篇。首篇講述VISICALC各種功能及控制使用方法，第二篇是舉出一個庫存管理之應用實例。

對於未曾接觸過VISICALC的讀者來說，閱讀完本文之後，當會令你對它有相當程度的認識，只要再加熟習，就能靈活運用這個非常強效的軟件。

對於對VISICALC早有認識的讀者，另一篇〔VISICALC的高級應用技術〕就非常會覺得有參考價值。

其他特別的功能

一、行寬的控制

VISICALC螢幕的標準的行寬是9格，亦即一螢幕可顯示四行，有時工作上並不須要這麼寬的空間，而有時則須更寬，此時便須改變每行的寬度。欲改變行寬只須鍵／GC，此時編輯欄會出現閃動的游標，表示要求我們輸入所欲希望每行的寬度，例如6或12等等，然後按下(R)即可。如要改回原來之型式（每行9格寬）只要鍵／GC 9(R)。當行寬改變時，有些數字或文字可能無法全部顯示，但VISICALC並未忘記原來的數字或文字，只要再將行變寬就可看到完整的數字或文字。但須注意印出報表時，是根據螢幕上所顯示內容的去列印，所以如果螢幕上每行是6格，則印出時也是6格的。

二、格式的控制

有時數字只須要整數部份，有時須要到小數點以後兩位（這是一般商業上對金額的表示法）。要使全部數字變爲整數，鍵／GFI，如果只想讓某一數字改爲整數顯示，則須將游標移至該數字，然後鍵／FI。欲使全部數字以小數點以下兩位顯示，則鍵／GF＄。如果只想使某一數成爲這種格式，則將游標置於該數，然後鍵／F＄。另外一般VISICALC之文字皆靠左，而數字靠右，有時不很好看，可鍵／GFR使全部靠右或／GFL使全部靠左，要恢復原來的樣子則鍵／GFG。

三、清除螢幕

鍵／CY。在由磁碟讀入新資料時，最好清除螢幕，以免新舊資料混雜。

四、行或列的搬移

想把一行或列移至他處時，先將游標移至該行或列，鍵／M，然後鍵入目標位置，最後按(R)即可。

五、固定標題欄名

有時因工作的表格太大，往往把標題欄名（前面例子中的SALES，COST等字）移到螢幕外而看不到，要避免此缺失，可將游標移到標題那一行或列，然後鍵／TC（若要固定列則鍵／TR）即可，欲解除固定則鍵／TN即可。

六、特殊函數

VISICALC還有一些有用的函數，分別介紹如下：

①最大最小值：用法和前面介紹的@SUM和@AVER—AGE相同，只是改爲鍵@MAX（……）（最大值），@MIN（…）（最小值）。括號中可爲一個範圍好A1、A10，也可以是個別的數字，例如@MAX（B1，B3，B7，A10），即找出B1，B3，B7，A10中最大的一數，另外也可將一段範圍和單個數字合用，中間用逗號分開，例如@MIN（A10，B1，B10，C11，C12）。

②查表函數：VISICALC中有兩個查表函數@CHOOSE及@LOOK UP。

③一般數學函數：如@SIN（），@COS（），@LN（）等，因用法完全和一般筆算時相同，不再敍述，唯在每一函數前須有@。

七、畫圖功能

VISICALC可畫簡單的長條圖，這裏舉一個簡單的例子示範。

先將螢幕清除，把游標移至B1，鍵／WV，把螢幕分成兩半，此時游標應在左半。在A1到A10內輸入1到10十個數字（可利用複製法）。然後鍵；使游標跳入右半螢幕，鍵／GC 20，使右邊行寬變爲20格，以便作圖。VISICALC只能畫長條圖，其畫法是將＊號塡入座標中，所須用的指令是1／F＊，當在某一座標鍵入此指令時，VISICALC會將此座標中的數字換成＊，例如在B1鍵入1，然後鍵／F＊，可以看到B1 中有一個＊，若此時鍵入5，則有5個＊出現，利用這種方法，VISICALC就可把數字變成圖形，現將游標移到B1，鍵＋A1(R)，則B1中有一＊，然後用複製法完成此圖：鍵／R(R)B2. B10 R(R)，螢幕應如圖七。

圖七

########VISICAL SAMPLE PROBLEM ########

MONTH	1.00	2.00	3.00	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00
SALES	200.00	220.00	242.00	266.20	292.82	322.10	354.31	389.74	428.72	471.59	518.75	570.62
COST	120.00	126.00	132.30	138.92	145.86	153.15	160.81	168.85	177.29	186.16	195.47	205.24
GROSS	80.00	94.00	109.70	127.29	146.96	168.95	193.50	220.89	251.42	285.43	323.28	365.38
SALE	130	143.00	157.30	173.03	190.33	209.37	230.30	253.33	278.67	306.53	337.19	370.91
COST	78.00	85.80	94.38	103.82	114.20	125.62	138.18	152.00	167.20	183.92	202.31	222.54
-GROSS-	52.00	57.20	62.92	69.21	76.13	83.75	92.12	101.33	111.47	122.61	134.87	148.36

可再按；使游標跳回左螢幕，試改變其中幾個值，看看右邊的圖如何變化。

由這例子可看出，VISICALC畫的圖大約只能畫30格，如果有一組數字是由10000到20000，則可以將它們各除1000，使這些數變成10到20之間，然後再用上面的方法畫成圖表。

庫存管理之應用範例

庫存管理之應用範例

現舉兩個例子是屬於小型商業應用之程式：其中一個是庫存管理，一個是出貨單據，兩個程式我們都造成表格的形式，應用時只須由磁碟中讀出空白的表格，填寫適當之數據即可，不必每次都造新的表格。這兩個程式聯合一起應用，也可以稱之爲庫存管理系統。

①庫存管理：庫存管理的形式有很多種，但性質都很相似，主要項目大致包括日期，貨品，貨源，進貨量，出貨量，庫存量等，有時再加上貨號及單據號碼等項目。

在這個例子中要建立的表格如圖A所示，各行由左至右分別爲日期貨源或買主，進貨量、出貨量、庫存量及備註欄，雙橫線以上是公司名，地址，然後是造表日期及造表時之庫存量。此表之功能是在填寫進出貨量以後自動計算庫存量，以及利用磁碟作爲儲存資料之介質，加以檔名系統化以後只要輸入檔名即可由電腦找到資料而不須用人工尋找。

現在來看看如何建立此一表格，以及如何輸入資料及儲存。

首先鍵／CY清除螢幕，依下面指示鍵入文字（後面寫A1：ITEM即表示在A1位置鍵入ITEM之意）：

A1：ITEM：　D1：INV：　F1：DATE：

這三位置是作爲整張表格的標題，ITEM：之後將來應鍵入貨品名稱，INV：之後鍵入第一次填表時之庫存量。D－ATE：之後鍵入第一次填表日期。然後在A2鍵／－＝〔R〕，接着鍵／R〔R〕B2.G2〔R〕，在第二列畫一雙橫線，接着輸入下列文字：

A4：DATE　B4：FROM／TO　D4：IN　E4：OUT　F4：INVENT。

然後在A5鍵／－－〔R〕，複製到G2：／R〔R〕B5，G5，最後鍵／GFR，使每格之文字靠左，以便以後和輸入之資料對齊。

此表之重點在庫存量之計算，目前之庫存量應等於上一庫存量加上進貨量或減去出貨量，因此先在F6輸入公式：＋E1＋D6－E6，E1爲開始之庫存量，D6爲進貨量，而E6爲出貨量，因此F6即爲現在的庫存量。現在F7輸入＋G6＋D7－E7，只要在E1填上庫存量，這表格就成爲一張能自動計算的庫存表。但目前這張表上只有兩格具有計算能力，如果設定一張表可填寫10格，則將F7之公式複製到F15即可：將游標移至F7，鍵／R〔R〕F8.F15〔R〕R－RR。最後三個R表示相對的複製，此時在E1輸入0，其他格皆空白，而G行全爲0，這就是一張可用的空白表格，將它存放於磁碟：鍵／SS BLANK〔R〕。以BLANK作檔名，表示這是空白表格，這表格最好存於VISICALC所規劃（FORMAT）的磁碟上。

要利用這張表時，先鍵／CY清除螢幕，然後鍵／SL BLANK〔R〕，就可叫出空白表格，貨品名稱鍵於B1及C1中。最初的庫存量鍵於E1，日期填於G1，進貨時，進貨日期由A6開始填寫，B6C6填貨源。數量填於D6，則F6的庫存量跟著就會自動算出。樣本如圖B所示。在圖A中增加了公司名稱及地址，這可利用插入空列的指令IR在A1上插入三個空白列再加以填寫，然後將整張表格當作空白表格存入磁碟。以後叫出表格填寫後以另一名稱存入，例如FORM1。當FORM1填滿時可用FORM1.F的名稱存入，以字尾的F表示表已填滿，此時就可再叫出空白表格填寫，將此檔名爲FORM2，使之成爲一有系統之資料。

當然一張庫存表可以不只有10格，而且也不只記錄一項貨品，要造較長之庫存表，只須將公式複製下去即可，也可以在一個螢幕上作兩三個以上的表格，同時記錄多種項目。

當表格很大時，每輸入一個新的資料VISICALC就會把整張表重算一次，以致於每鍵一個數字要等兩三秒以上，要避免這項困擾可在輸入資料前先鍵／GRM，這個指令使－VISICALC停止自動計算，如此可以較迅速輸入大批資料，等資料輸入完畢再鍵／GRA恢復自動計算能力。

注意輸入日期時須先鍵"，以免VISICALC將5／1當作5除以1。

圖A

```
          COMPUTING      AGE      LTD.
                   3-312007
---------------------------------------------------
Item                    INV
                        DATE:
---------------------------------------------------
Date    FROM / TO      IN      OUT    INVENT
---------------------------------------------------
```

圖B

```
          COMPUTING      AGE      LTD.
                   3-312007
-------------------------------------------
Item    NOTOBOOK        INV          1000
                        DATE:         5/1
-------------------------------------------
Date    FROM / TO      IN      OUT    INVENT
-------------------------------------------
5/1     SMITH CO       500            1500
5/2     ITOP CO                400    1100
5/3     POP CO        1000            2100
```

圖C

```
            COMPUTING      AGE         LTD.
                    3-312007
-------------------------------------------------
TO   :  GENERAL   CO.        INV          1000
DATE:     5/11/84            DATE:         5/1
-------------------------------------------------
ITEM    DESCRIPT. QUANTITY  UNIT PRICE   AMOUNT
-------------------------------------------------
        105NOTEBOOK      50      4.5        225
        108BLUE INK      20      7.5        150
        104PEN          100      8.5        850
-------------------------------------------------
REMARKS:
                         TOTAL   =         1225
```

庫存管理之應用範例

②出貨單據：一般出貨單要塡寫的部份包括貨號、品名、數量、單價、金額及總計等項目，本範例所建立的出貨單特色是只須塡寫貨號、品名及數量，程式將會自動查出貨品之單價，算出金額及總計。

要作到這一點，首先須告訴電腦各種貨品之單價，亦即先輸入一價目表，作爲空白表格之一部分，當輸入貨號時便利用@LOOK UP查表指令去找出單價，然後依據單價算出各貨品之金額，最後將所有金額相加而得到總計。

作好的表格並不將價目表印出，印出之表如圖C所示。先畫出表格之雛型。公司名及電話：B1：COMPUTING C1：AGE D1：LTD　B2：" 3-TEL C2：312007

然後在A4鍵T0：表示此處塡受買人。A5鍵DATE：作爲塡寫日期之處。另外若須塡寫發票號碼也可自行再定出空間。在A6鍵／—— 〔R〕然後複製到F6。接著塡入下列項目：

A7：ITEM＃　B7：DESCRIPTI　C7：ON
D7：QUANTITY　E7：UNIT PR.　F7：AMOUNT

在A18到F18再畫一條線，A19鍵REMARK，E19鍵TOTAL，此時整張表應如圖C一所示。

現在輸入一張價目表，假如有10種貨品，編號由101到110，現在H1到H10分別鍵入101到110十個數，另外在I1到I10分別鍵入0，100，50，30，45，50，70，80，35，15，代表各貨品之單價。

我們希望當在第A行塡入貨品編號時，程式能從上面的表中查出貨品的單價而自動將單價塡入單價欄（UNIT PR.），這項工作可由@LOOK UP指令完成，此指令之形式爲@LOOK UP（V，RANGE）V代表所要查的資料，在本例中即爲貨品編號，而RANGE代表一表格之位置，在本例中是價目表H1至H10（H1.H10）。例如先在E8鍵@LOOK UP（A8，H1.H10）〔R〕，然後在A8鍵入105，這時在E8的@LOOK UP指令就到H1到H10中去找105，找到105後把105右邊的45放到E8位置，而完成查表動作。由上面的說明可推知E9的指令應爲@LOOK UP（A9，H1.H10），可以用複製法完成其餘部份：在A8鍵／R〔R〕A9.A17〔R〕RNN。因查表之範圍永遠爲H1到H10，所以複製時後兩項以絕對方式複製。

上面完成了自動塡寫單價的功能，而金額自然是數量乘上單價了，例如F8之公式應爲＋D8＊E8，然後可以用複製法作到F17，都以相對的方式複製。

最後F19的總計就是由F8累加至F17，即我們以前用過的@SUM（F8.F17）〔R〕。

這樣便完成了我們的出貨單，當然在價目表中的貨號和單價是視貨品而定，但須注意的是貨品編號須由小而大向下排列（或由小而大由左向右作橫的表），不可無規則的亂排，否則在查表時會有錯誤產生。

這張表同樣以一名稱存入磁碟，要塡表時叫出即可續項塡寫，此表須塡的部份有受買人、日期、貨品編號、貨品名、數量等項，要注意的仍是鍵入日期時先鍵"。各項目塡好時就可將表印出，例如在本例，將游標移至A1鍵／PPG25〔R〕〔R〕，即可印出單據。

至此，相信大家對VISICALC用於商業寫字樓用途上的偉大貢獻。希望各位能好好善用。■

更正

第十一期第65頁〔告訴我〕

程式行句390應改爲 `HGR:HCOLOR=3 FOR X=.....`

第43頁〔空SECTOR 圖表〕

程式行句80應改爲：`NA$="FREE SECTOR BAR FOR "+NA$`

行句210應改爲：`HTAB 2:PRINT A$;A$;LEFT$(A$,3)`

行句350應改爲：`IF A<(2^T)THEN PRINT"*"`

第67頁〔趣意盈盈的歡樂棕〕JOYSTICK MENU程式

行句150 應改爲
```
INVERSE : VTAB 4 + OP: HTAB
10: PRINT OP;"..........";P
ROG$(OP): NORMAL
```

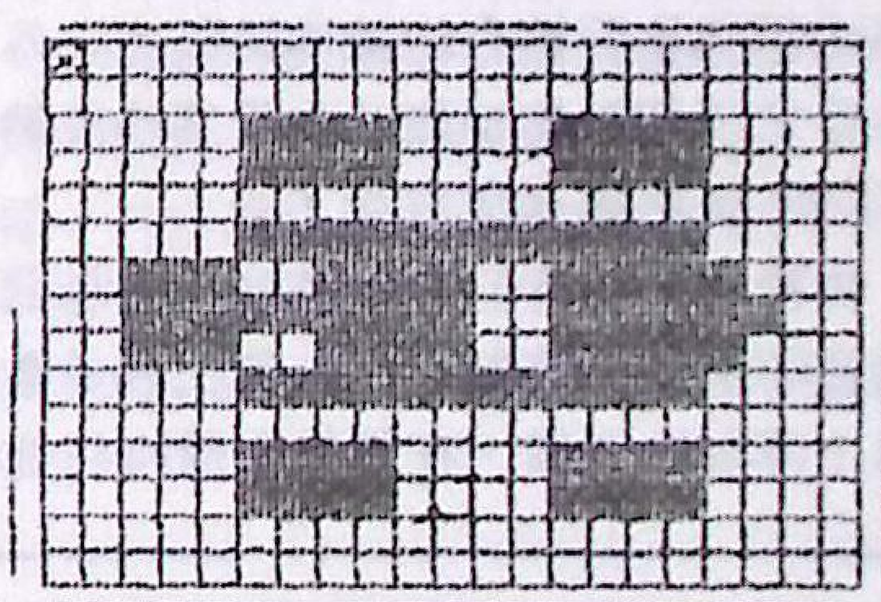
行句340 應改爲
```
VTAB 2: HTAB 5: INVERSE : PRINT
" JOYSTICK MENU BY COMPUTING
AGE": NORMAL
```

行句360 應改爲
```
VTAB 4 + N: HTAB 10: PRINT N
;"...........";PROG$(N)
```

程式JOYSTICK 1行句80應改爲
```
IF  PEEK ( - 16286) > 127 THEN
 PRINT  CHR$ (13) +  CHR$ (4
);"RUN JOYSTICK MENU "
```

第七期〔蘋果杯大賽車〕漏登圖(1)，補登如下：

圖(1)



程式JOYSTICK 2行句130應改爲
```
IF  PEEK ( - 16286) > 127 THEN
 PRINT  CHR$ (13) +  CHR$ (4
);"RUN JOYSTICK MENU "
```

程式JOYSTICK 3行句160應改爲
```
IF  PEEK (49152) = 155 THEN
 PRINT  CHR$ (13) +  CHR$ (4
);"RUN JOYSTICK MENU "
```

程式JOY TONE行句110應改爲
```
PRINT  CHR$ (13) +  CHR$ (4)
;"RUN JOYSTICK MENU "
```

專業級文字處理器 WORDSTAR

黃國亮

市面上有着多種的文字處理器出售，較流行的有APP—LE WRITE Ⅱ，MAGIC WINDOW Ⅱ及WORDSTAR等，在這些文字處理器當中，WORDSTAR可說是最具威力。在AP Ⅱ上使用WORDSTAR最少要具備48K RAM、Z80咭、80COLUMN咭及一部磁碟機。當然，處理較大的檔案時，一張16K RAM 咭或多一部磁碟機會帶來不少方便。

由於WORDSTAR是一個頗爲複雜的文字處理器，初學者是需要一段時期才可完全利用它的各種功能的，不過，這並不等於它不適用於他們，相反地，初學者却可以在極短時間內便可享用到它的許多好處，如果你曾使用過其他文字處理器如MAGIC WINDOW Ⅱ 等，相信你在使用時都需要經常翻閱它的使用手冊，因爲你是絕不可能在短時間內弄清楚各個按鍵的功能，而WORDSTAR就爲初學者提供了莫大的方便，這是由於它的所有基本鍵盤上的功能都顯示在螢幕的MENU 上，使用者沒有必要強記按鍵的功能，而在使用過程中漸漸的將各個按鍵功能熟習起來，更可以按照自己對按鍵功能的熟習程度更改HELP LEVEL，使WORDSTAR更配合使用者的操作。

由於WORDSTAR的功能可說多得不可勝數，在這裏不作詳談了。它除了擁有一般文字處理器的基本功能外，它更可使用點指令（DOT COMMAND）及一些編印控制碼（PRINT CONTROL CHARACTERS）來達到完美的文件印製格式及效果。由於WORDSTAR能利用設置（INSTA—LL）過程來配合不同的硬件組合，所以它是絕對能配合你所擁有的印字機的。除了點陣式印字機外，WORDSTAR更可配合DAISY WHEEL 印字機，加上WORDSTAR 的MI—CROJUSTIFICATION 功能，效果更是完美。除了一些WORDSTAR所擁有的基本編印功能外，使用者更可以在設置時自定四個鍵盤功能及改變某些按鍵功能，這樣WORD—STAR 便更適合使用者了。

由於在設置時已將各按鍵的功能選定，使用者就沒有必要知道印字機的控制碼了，現在大多數點陣式印字機都有幾種以上不同的字體，在轉換字體時往往都需要翻查印字機使用手冊，找出它的控制碼，例如使用MAGIC WINDOW時想轉換字體，就必須先打入CTRL —B，後再打入印字機的控制碼去轉換字體。在使用WORDSTAR時，你可不必理會這些控制碼，你只須在CTRL —P後再按入一個已設置的功能鍵便完成了字體轉換的工作了。

除了以上各種功能外，WORDSTAR更可配合其他軟件使用，例如配合MAIL MERGE，可使信件的打印處理工作達到最高效率，文件更可以利用SPELL STAR 去爲你找尋錯字，這相信是大意打錯字者的恩物了。（請參閱第十一期電子辭典的典範）

有點值得一提的便是WORDSTAR的設置過程（INSTA—LLATION PROCEDURE）了，設置過程在WORDSTAR的INSTALLATION MENUAL已說得頗詳細，不過對於初接觸電腦的人會感到一定的困難。在設置過程中，在PRIN—TER MENU 中我感覺選擇A的ANY "TELETYPE—LIKE" PRINTER會比C的 "TELETYPE—LIKE" PRI—NTER THAT CAN BACKSPACE 在印刷時有較流暢的效果（我使用的是EPSON RX—80 F／T），而選擇PRIN—TER DRIVER 時最好是選擇PORT DRIVER，大家都知道在使用GRAPPLER十咭作PRINTER INTERFACE時，作爲CP／M "LIST" DEVICE 的印字機在印字時是會弄得80 CULUMN 的畫面閃動至不能觀看，但是如果利用PO—RT DRIVER 作輸出，不但速度有改善，在編印文件時，80 CULUMN 的畫面是一樣的穩定清楚的，這對於WORDSTAR是非常重要的，因爲WORDSTAR能夠在沒有PRIN—TER BUFFER 下，一方面印製文件，一方面可同時進行文字處理的工作，使用者再不需要苦候文件編印的完成才可繼續進行他的文字處理工作，這省回不少時間。

如果想要重新訂立那些鍵盤功能，這便需要進行一次—NORMAL FIRST—TIME INSTALLATION了，使用者必須將控制碼打入適當的地址（MEMORY ADDRESS）內，按鍵的控制地址可參閱INSTALLATION MANUAL 所提供的USER LISTING。■

如有疑問，請電3-312007 編輯部查詢

投稿規則

（一）本刊絕對公開，歡迎任何形式的投稿，包括：軟件程式，電腦硬件 REVIEW 及製作，商用、遊戲軟件 REVIEW，技巧使用，各種電腦語言介紹詳述等等。
（二）來稿以原稿紙單面橫寫爲合。
（三）軟件程式投稿，如程式有一定長度，請附同磁碟或磁帶一併寄來，程式必須加以清楚說明、用法及原理。
（四）寄來之磁碟或磁帶於稿件發表後即退回。
（五）稿件字數長短不限。
（六）請勿一稿兩投。
（七）所有投稿必須附上作者中英文眞實姓名、通信地址及電話（方便電腦紀錄），發表時筆名隨意。
（八）來稿可寄九龍中央郵局信箱 71193 號電腦時代收，信封面請註明〔投稿〕，或親身携來本刊。



卓越的繪圖工具
E-Z DRAW

朱永康

E—Z DRAW是一個多姿多采的程式，它帶給繪圖者無窮的方便，也使得繪畫圖案變得輕易。用它來繪畫三角形、四邊形、圓形、橢圓形……可謂輕而易舉。

簡單操作

十字形的CURSOR 以一組鍵（I、J、K、M）來控制它上下左右移動，每按一下鍵，它所移動的點數有1－9可供選擇（按1鍵表示移動一格……）。要在螢光幕上繪出一點，只需按SPACE BAR 便可以。按 F鍵便會看到螢光幕下面出現幾行字——X、Y所指的分別是CURSOR 的橫、縱坐標值；X0、Y0所指的坐標值乃是最後一個繪點的所在。

要繪出一個長方形，只需在所想的長方形的左上角留下一點，然後將游標移至右下角，按下R鍵，一個完整的長方形便會立刻出現。

繪畫三角形只需留下兩點以確定三角形的兩個角，然後游標移到第三個角上，按下〝T〞，三角形便順利繪好了。

平行四邊形的畫法較特別，你必須定下兩點以確立它的兩個角，然後將游標移到第三個角上，按下〝P〞，你所作的平行四邊形便會立刻出現。

要繪畫圓形，首先要在圓心處繪出一點，並將游標移至圓周的任何一個點上，按下〝O〞。要繪畫橢圓形，必需定下它的兩個圓心，然後將游標放在邊上，按下〝E〞。

顏色控制

也許你會嫌現在所畫的圖形只是白色，不夠鮮艷。不用担心，E—Z DRAW提供了8種顏色選擇，按下CTRL—C，然後按所需顏色的編號（0－7）。有時你會想到將自己所畫的三角形、圓形填上顏色，這目的非常容易達到，例如：平常你要畫三角形的時候要按下〝T〞，現在改爲按〝CTRL—T〞，你的三角形便會塡滿你所選用的顏色，其他的圖形也是如此——〝P〞改爲〝CTRL—P〞；〝E〞改爲〝CTRL—E〞……

如欲在螢光幕上印出文字，可用@（SHIFT—P）達到此目的，文字字形的大小，由以下幾個鍵來控制：

CTRL—R　正常字形大小
CTRL—E　較大的字形
CTRL—T　窄身字形
CTRL—W　扁形的字

如希望在敲鍵的時候發出聲音，可按CTRL—V。無論大小字形，皆可選用顏色，方法也是按CTRL—Cn（n是由0－9的任何一個鍵），以下是各種顏色的對應編號：

0——白色
1——綠色
2——紫色
3——橙色
4——藍色
＊5——黃色
＊6——紫藍色
＊7——粉紅色
＊8——水色
9——黑色

上面帶有＊號的顏色只可在大字形的時候才適用。（即是由CTRL—E決定的字形。）基於E—Z DRAW有多種不同大小的字形與顏色，將文字顯現在螢光幕上便簡單得多了。（附：如欲轉回繪圖模式，可再按@一次）

簡單繪點

使用E—Z DRAW的時候，你可能會覺得每按一下SPACE BAR留下一點太麻煩，不用担心，只消按〝／〞鍵，你每移動CURSOR 一步，便會留下連續的點與線。要回復原狀，可按〝．〞。有時，當你繪點或作垂直線的時候，你會發覺電腦沒有把你所作的點或線顯現出來，不要以爲E—Z DRAW 發生錯誤，這是基於你所選用的顏色與螢光幕的特殊關係，倘若使用黑、白色，此情形便不會再出現。（關於螢光幕與顏色的關係，可參看任何一本有關GRAPHICS的書籍）

要解上述的困難，可將CURSOR 移左或移右一點，此現象便不會再發生了。

輸入輸出

要將所畫的圖案放入磁碟中，可按S（SAVE）鍵，之後，你便會看到磁碟內檔的名稱，然後是以下幾行字：

S(AVE) PICTURE OR IMAGE
(OR <CR> TO ABORT)
WHICH WOULD YOU LIKE ?

PICTURE 是將整個螢光幕的資料寄貯到磁碟上。至於IMAGE 則是將螢光幕的某一部份移到磁碟上，那一部份，則由你自己決定，你只要在左上角留下一點，並將CURSOR移到對角點上，按下S和I，並輸入IMAGE的名稱，在那兩點內的一切便會送到磁碟上。要由磁碟上提取PICTURE 或IMAGE，則按L和I或P，然後輸入檔名。欲運用2號磁碟機，可按SHIFT—2（SHIFT—1則指一號磁碟機）。

SHIFT- N

當你按下SHIFT—N，便會看到以下這幾行字：

```
A) MIRROR IMAGE
B) UPSIDE DOWN
C) SLANT RIGTH
D) SLANT LEFT
E) COMPRESS HEIGHT
F) EXPAND HEIGHT
G) COMPRESS WIDTH
H) EXPAND WIDTH
I) ROTATE 90 DEGREE
J) SAVE P.1 ON P.2
K) RESTORE P.1 FROM P.2
```

要使用以上各個特殊指令，首先必須定一個區域來執行，這個區域的委定方法，與繪長方形的方法一樣，定下二個對角點，然後按SHIFT—N，你便可以選擇由A—K各個鍵的特殊功能，以下便是這些鍵的簡單陳述：

A——將區域內的一切自左至右，自右至左地倒轉過來。

B——將整個區域自上至下地倒轉過來。

C——它將整個區域內的一切往右斜，如區域內有一個正方形，執行後便成爲一個平行四邊形。

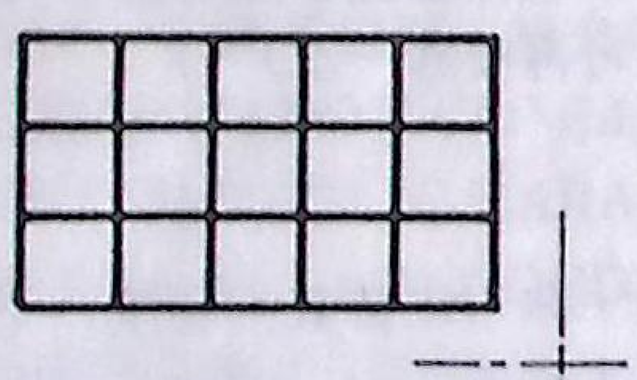


圖 (A)

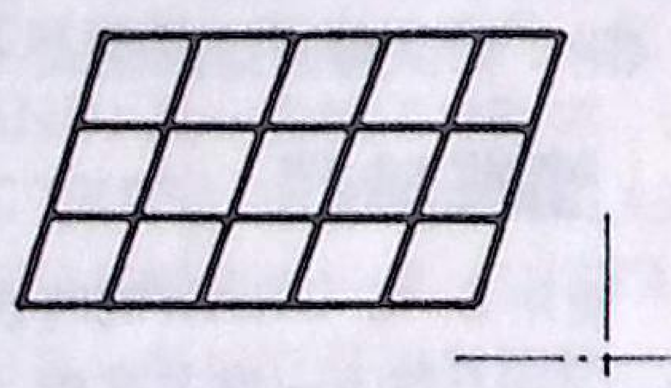


圖 (B)

圖(A)是執行前的樣子

圖(B)是執行後的樣子

D——與C剛好相反，圖(C)是圖(A)執行"D"指令後的樣子：

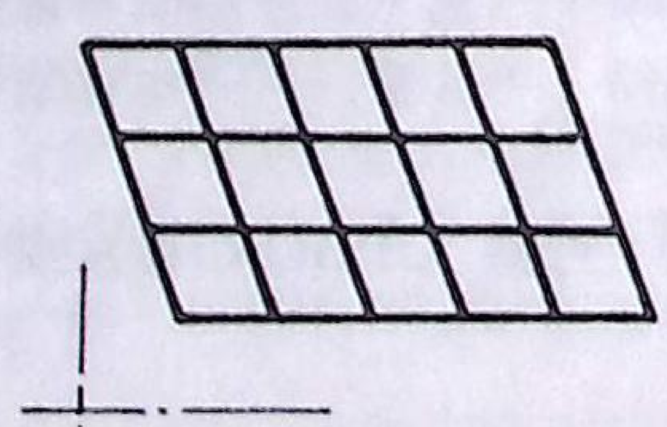


圖 (C)

E——將整個區域壓扁。

F——與E相反，將區域拉高。

G——將整個區域縮窄。

H——與G相反，將區域拉濶。

I——將區域內的一切轉90度。使用此鍵必須首先作出三點，區域轉動後所擺放的位置由第一點所決定，第一點所在的地方，將是區域轉動後的左上角的位置。

下圖(D)是執行前的樣子，點旁的數字是作點的先後次序：

圖(E)則是執行後的樣子。

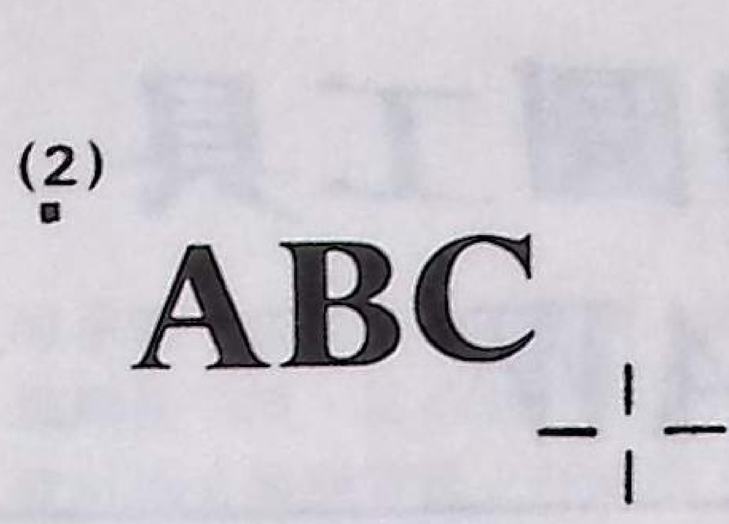


圖 (D)

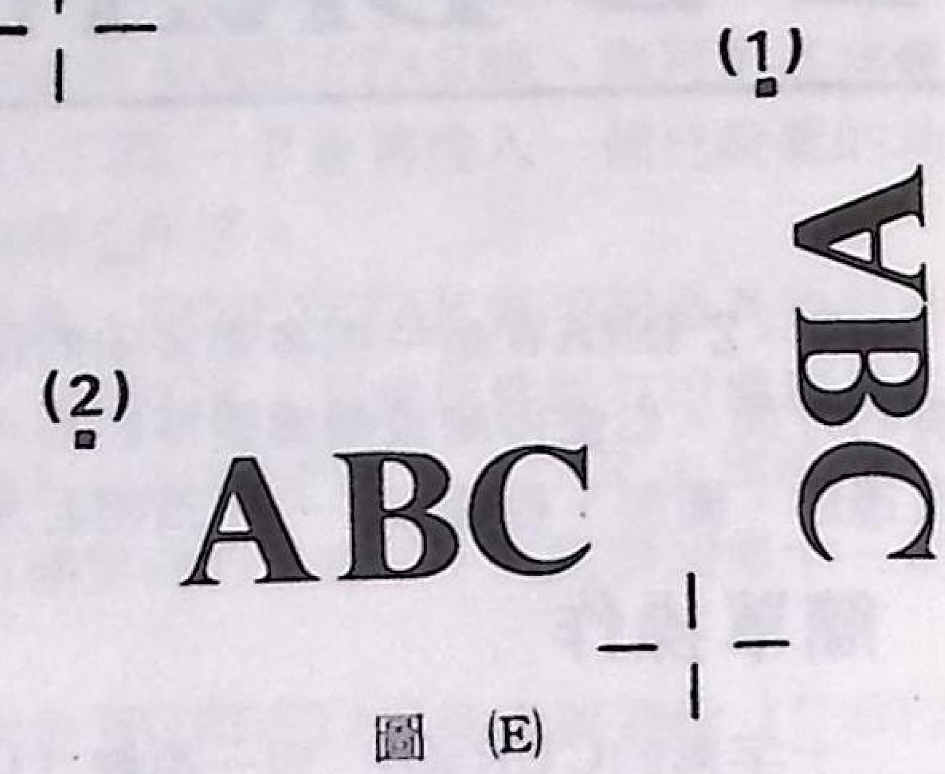


圖 (E)

J——此鍵比較特別，它不須預先確立一個區域，它只是將第一頁內的一切移至第二頁，以第二頁作爲一個緩衝區。此功能非常有用，如果你在繪圖中將資料移至第二頁，就不怕你自己辛苦繪畫的圖案因一時的不小心而受到損壞了。

K——與J正好相反，它將第二頁內的一切移到第一頁上，即使第一頁內的圖案已受到嚴重的破壞，你只需要執行這個功能鍵，電腦便會由第二頁緩衝區內提回已貯下的圖案。

CTRL- B

本繪圖模式時，你按下CTRL—B，便會看到以下這行字：

ARE YOU SURE？（Y）

它的功能是將整個螢光幕變爲現在所採用的顏色，譬如現在所用的是黑色，按下CTRL—B，然後按Y鍵，整個螢光幕的一切都會被清除掉。

當在文字模式時，按下CTRL—Bn（n是由0—7的任何一個數字）它的功能卻完全相異，其列述如下：

0——與第一頁的底色作EOR。
1——與第一頁作OR。
2——與第一頁作AND。
3——與第一頁作EOR。
4——與第二頁的底色作EOR。
5——與第二頁作OR。
6——與第二頁作AND。
7——與第二頁作EOR。

關於EOR、OR、AND的作用，可參看一些關於高級繪畫技巧的書籍。EOR、AND、OR的功用十分特別，它與底色關係十分密切，讀者不妨多練習使用它們，只要多練習，多想，不難發覺它們的奧妙之處！（例如要保持原有的圖案，又想繪出新的圖案或文字在上面，就必須使用它們了！）

其實，E—Z DRAW本身是一個非常有用的繪圖程式，不獨可在螢光幕上繪製圖案，而且還可與PRINTER連接，更有多種字形可供用家選擇（檔名後連有"F"的就是字形，它可以由LOAD FONT輸入電腦內）。總之，多使用E—Z DRAW帮助繪圖，你便會發現它的奧妙之處！■

CP/M 55個公用程式——POWER(上篇)

POWER是一套CP／M的綜合性公用程式系統，它主要是用來作記憶體及磁碟的修護之用。

以往CP／M的程式磁碟一旦將一個檔案刪除之後，便無法恢復，可不像DOS的磁碟擁有那麽多的修護工具，可以很輕易地將之UNDELETE。另一方面經常困擾我們的問題是磁碟上出現BAD SECTOR令一些檔案無法恢復時，如何修護確成一個大問題，就單這問題來說，POWER已十分值得推荐。

POWER一共有五十五個功能，它的設計給予CP／M用家極大的彈性，例如提供目錄號碼以至一切的程式操作都無需按入冗長的程式名字，而且可以提供一些很好的護理功能，例如你操作一部未插入磁碟的磁碟機時，POWER會出現"CAN'T LOG ON A:"的訊息而不引致記憶體被毁。

POWER所擁有的55項功能

①CHECK：這項功能是用來讀取一個磁碟上的檔案，之後POWER會將輸出一個CHECKSUM到該檔案。

②COMPARE：可以將兩段位置的記憶體內容作比較，如同MONITOR內的VERIFY功能。

③COPY：將磁碟上的檔案由一部磁碟機寫去另一部磁碟機上，採用數字目錄而不需打入太長的程式名稱。

④DIR：可將磁碟上的檔案列表出來。

⑤DISK：將磁碟上各項參數及格式化的資料顯示（圖1）。

⑥DS：將記憶體內容傾印，並可以更改它們，輸入方式可以爲十六進制數碼，ASCⅡ字元，二進制或十進制模式，可以說是DDT的改良。

⑦DUMP：可將記憶體的內容以ASCⅡ代碼傾印出來。

⑧DUMPA：能將記憶體內的一個區域的內容以ASCⅡ代碼傾印出來。

⑨DUMPH：能將記憶體內的一個區域的內容以十六進制代碼傾印出來。

⑩DUMPX：能將記憶體內的內容以十六進制碼及ASCⅡ碼印出。

⑪ERA命令：可將檔案刪除，它相當於DOS的DELETE命令。

⑫EX命令：是EXECUTE的縮寫，它可以執行指定位址開始的記憶體內的程式，完成後會回到POWER去。

⑬EXIT命令：將會離開POWER而跳往用家指定的位址去，若無指定位址則會跳往$0000去，這是CP／M WARM BOOT的開始位址。

⑭FILL命令：將會以指定的十六進制BYTE來塡入指定的記憶區域之內。

⑮GO命令：能載入並執行記憶體內的程式。

```
XPLEASE TYPE THE NUMBER OF YOUR CHOICE
A0>DISK
disk capacity:        140K
tracks:                38      3 system
sectors/track:        32     32 last
sectors/system:       96     12 dir
dir entries:          48      1K
sectors/group:         8      1K   8BH groups
kbytes/extent:        16K

A0>DISK
disk capacity:        140K
tracks:                38      3 system
sectors/track:        32     32 last
sectors/system:       96     12 dir
dir entries:          48      1K
sectors/group:         8      1K   8BH groups
kbytes/extent:        16K
```

圖一

⑯GROUP命令：可以依據CP／M內定格式而將一組檔案編組。

⑰JUMP：跳往指定位址執行程式，之後會WARM BOOT CP／M。

⑱LOAD：能將磁碟內檔案載入到記憶體內的任何位址上。

⑲LOG：將POWER的自設値顯示。

⑳MOVE：將記憶內一個區域的內容移往其它指定位址。

㉑PASS：是PASSWORD的縮寫，它會啓動一些保護功能並保護使用的記憶區。

㉒READ（SCREEN）：將磁碟上任意磁軌上的內容直接讀入並顯示在螢幕上。

㉓READ（MEMORY）：將磁碟上任意磁軌及磁區讀入到指定的記憶位址上。

㉔READGR（SCREEN）：將任何CP／M檔案組讀入並在螢幕上顯示。

㉕READGR（MEMORY）：將任何CP／M檔案組讀入並在任何指定的位址上去。

㉖RECLAIM：相當於UNDELETE，可將一些被删除的檔案恢復。

㉗REN：是RENAME的縮寫，可以將檔案的名字更改。

㉘RESET：能夠不使用CTRL—C而將記憶體內存放的磁碟機資料更新。

㉙RUN：從數字目錄上選擇並執行程式。

㉚SAVE：能將記憶體的任何位址的內容貯存到磁碟上。

鑑於POWER的功能太多（總共有五十五項），篇幅所限，未能一次全部介紹，有關餘下的廿五項功能及POWER整體的評語，筆者下期再談。■

ENTERTAINMENT

最新遊戲軟件評價

李德華

花生漫畫 SNOPPY

遊戲名稱： SNOPPY

遊戲形式：反應及要數學頭腦

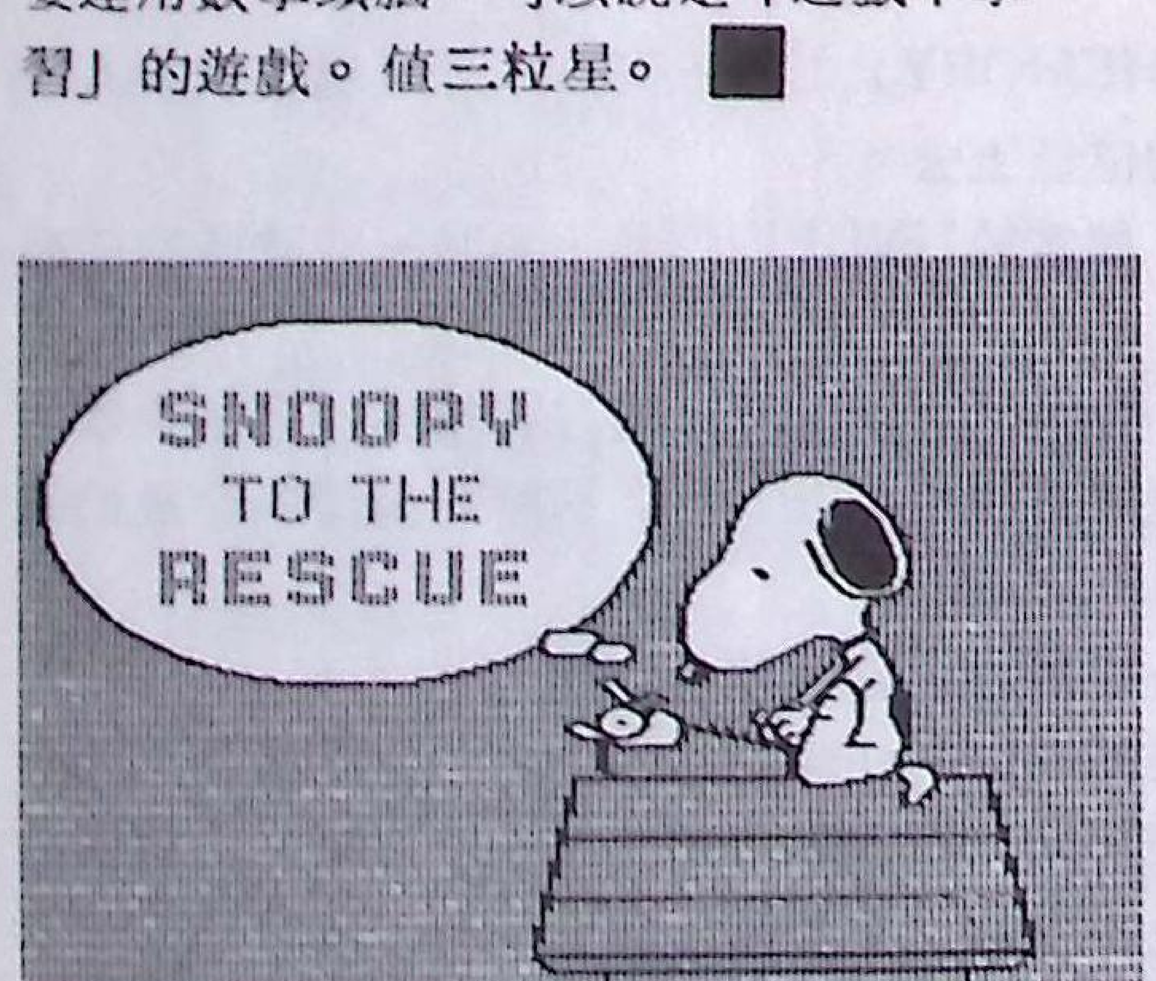
評分：＊＊＊

相信各位讀者一定看過花生漫畫，而對史洛比這隻頑皮狗一定耳熟能詳了。

而今天爲各位所介紹的遊戲就是SNOPPY，在未說這個遊戲前，先聲明香港有兩個版本，一是有BUG，另一是運行無阻，但相信各讀者所擁有的版本必是後者。現在言歸正傳介紹SNOPPY，筆者認爲封面設計非常美麗，最特別是SNOPPY在沉思及打字的一段，表情非常有趣，這處値得一讚。

遊戲的目的是把正在轉動的數字（一共五個）相加，而等於畫面上的數字，例如數字是12，便要把數字移到3、2、3、4、0，這樣已經成功了第一步，跟着行上最左面的建築物，便大功告成了，途中會遇到不少怪物，但只要吃了骨頭，便能夠把怪物消滅。而其他LEVEL也大同小異，只是形式有所轉變，而筆者認爲最難玩是LEVEL 5及LEVEL 6，這兩個LEVEL更加上一些特別的東西（第五排是橋，而第六排是神燈），當你接觸到它們的時候，便能夠彈起，相反不小心便會墜地身亡，而筆者教大家不要心急。第五排要看到「橋」在中間的時候，便跳起，到最高點的時候，再跳起，便能到達你想到的地方。而第六排大同小異（只是換了神燈），最主要是不要撞到最上層的「橋」。

以近期來說，這個遊戲可算是構圖最美麗的一個，它不但要手急眼快，還要運用數學頭腦，可以說是「遊戲中學習」的遊戲。値三粒星。■





超時空要塞

遊戲形式：射擊

遊戲名稱：超時空要塞

評　　分：＊

各位讀者一定欣賞過「超時空要塞」這套卡通片，而一定對阿寶這個人物念念不忘。

筆者最近發現商場出現「超時空要塞」這個遊戲，筆者由於被遊戲的封面所吸引，於是便拷貝了一個。但當我開始玩的時候，便發覺這個遊戲不是我想像中緊張刺激和畫面生動，而是些低級趣味的玩意。

遊戲的目的是把襲擊你的太空船消滅，而你的武器是「弧形飛彈」，每次可以發射五粒。雖然這個遊戲畫面非常單調和缺乏娛樂性，但亦有它的難度性。大約過了十秒攻擊後，便升LEVEL。整個遊戲來說，設計太過簡單，只適宜一些不愛動腦筋的朋友們玩耍，而對於愛動腦筋的人仕，這個遊戲似乎不大適合他們了。

經過筆者分析後，這個遊戲有少許抄襲成份。我懷疑這個遊戲是本地的作品，斷不會是美國的作品，筆者不是彈我們香港人的作品，而本地也曾經推出過BURGER TIME（即漢堡飽），這個是本地人較好的作品。

總括來說，這個遊戲是近期來較差的一個。■

DARK CRYSTAL 2

相信很多讀者都玩過DARK CRYSTAL，而DARK CRYSTAL ］［大致上差不多，仍然是要找尋魔水晶，把它由黑暗變回光亮。但這遊戲只需一張磁碟，而以前的就要幾張磁碟了，而且非常多難關。魔水晶2號不單只比以前容易玩，而且還有選擇，免得要傷腦筋。你開始玩的時候，有些難關和圖像已似曾相識，所以有些難關已不用再「過」了。當你去到一座玻璃山的山脚時，問題有幾個答案，其中一個答案是要爬過這座山，電腦便會發出轉去第二面碟的命令，而繼續去尋找魔水晶了，請記着這個遊戲有不少陷阱，行錯一步便足以從頭玩過，請記着！筆者用了三日時間來破解魔水晶2號，而我總共用了七十一步去完成這個遊戲，但魔水晶2號的作者用了七十二步，恰巧慢了筆者一步。

編者註：本文作者用了71步去完成破解魔水晶2號，該71步行法刊登於一COMPUSOFT 磁碟月刊創刊號。讀者留意。■



電子郵箱

用家朋友可利用電腦設備彼此通訊傳達消息

相信大部份讀者都擁有一個或以上的文字處理器軟件。對於擁有印字機的人來說，文字處理器無疑是十分方便，因爲一切的文書紀錄私人通訊可以保存並於有需要時重印副本。不過無論功能怎樣強大，假如沒有印字機，便令人一籌莫展了。

讀者之中，許多時可能要與電腦用家朋友彼此通訊，如果有印字機當然方便，可以利用文字處理器，發揮所擁有的電腦設備功能。那麼，沒有印字機又怎辦？

今期主要就是爲大家介紹一個電子郵箱的應用軟件。透過這個電子郵箱，各位可以將各項的訊息告訴朋友，或彼此互相通訊。

這個電子郵箱在使用上非常方便。當程式執行時，出現的主功能表會有七項選擇。

（1）載入並顯示文本檔案
（2）在一檔案上加添新訊息
（3）增加一個新檔案
（4）更改檔案
（5）刪除舊檔案
（6）觀看檔案名稱
（7）離開本系統

當第一次執行電子郵箱時，需使用功能（3），開設新檔案並寫入一些訊息。

第一項功能可以在以後將檔案讀入並觀看，你只要鍵入檔案名稱便可以了，假如忘記了名字的話，本系統的第6項功能可以將這些檔案的名稱顯示給你們；返回MENU可以按CONTROL－C。

第二項功能可以在一個原有的檔案上加添資料內容，它主要是利用DOS的APPEND功能來達成。不過無論我們使用第二或第三項功能，所輸入的通訊長度都要少於224個字符，這是因爲緩衝器的空間問題，又檔案名稱的長度要少於10個字元。

第五項功能可以將磁碟上的檔案刪除，若果檔案名稱是正確的話，你又按入DEL來繼續時，檔案便會被刪除。第六及第七功能相當簡單，不用多解釋。

☆程式的工作原理

本程式最具興趣的是由8000行－8070行的輸入副程式，它直接使用鍵盤緩衝器，因此可以輸入任何字元，甚至CONTROL字元在內。任何時候從鍵盤上輸入時，字元會貯存在＄200至＄2FF之間的緩衝器之內，因此我們採用MONITOR內的GETLN副程式截取緩衝器內的資料，假如按下RETURN鍵，GETLN副程式會停止工作而回到BASIC去，第8020行字元會貯存在字串變數ST＄之內，若果RETURN鍵被按下時，程式脫離GETLN到3090行使ME＄（MC）的內容和ST＄的內容相同。

在完成訊息輸入之後，程式會問你日期，格式是MM／DD／YY。這時你可以輸入日期或按RETURN掠過。本程式可以最多在一個檔案上容納多達100項訊息。

每次當我們選擇第一項功能時，並且選擇要觀看的檔案名字之後，便會由一個相當強效的輸出格式網路來執行，由1000行開始。你可以應用它作其它用途，只要先設定LM（LEFT MARGIN）及RM（RIGHT MARGIN），不過RM必須在螢幕範圍之內，本程式的RM爲32。

其它的副程式功能十分簡單，留待各位去研究了，不過以後在新版本推出時，筆者會再和各位探討。

```
]PR#0
]LIST

1  REM  ********************
2  REM  *       .          *
3  REM  *    ELECTRONIC    *
4  REM  *    MAIL - BOX    *
5  REM  *                  *
6  REM  * BY COMPUTING AGE *
7  REM  *                  *
8  REM  *  COPYRIGHT 1984  *
9  REM  *                  *
10  REM ********************
100  POKE 768,104: POKE 769,168: POKE
     770,104: POKE 771,166: POKE
     772,223: POKE 773,154: POKE
     774,72: POKE 775,152: POKE 7
     76,72: POKE 777,96
110  ONERR  GOTO 32000
120  DIM TI$(100),ME$(100),MD$(10
     0)
130  TEXT : HOME : INVERSE : PRINT
     "+++++++": FOR I = 1 TO 22: PRINT
     "+      +": NEXT : PRINT "+++
     ++++";: VTAB 1: VTAB 8: HTAB
     2: PRINT "+++++": VTAB 10: HTAB
     3: PRINT "EMB": VTAB 12: HTAB
     3: PRINT "BY:": VTAB 14: HTAB
     2: PRINT "COMPU": VTAB 16: HTAB
     2: PRINT "++++++": VTAB 1: FOR
     I = 8 TO 40: HTAB I: PRINT "
     +";: NEXT : FOR I = 2 TO 23:
      VTAB I: HTAB 40: PRINT "+";
     : NEXT : VTAB 24: FOR I = 8 TO
     39: HTAB I: PRINT "+";: NEXT
     : POKE 2039,43: NORMAL : POKE
     32,7: POKE 33,32: POKE 34,1:
      POKE 35,23: HOME
200  VTAB 4:A$ = "<<MAIN MENU>>":
      GOSUB 11000: VTAB 22:A$ = "
     (C) 1984 COMPU HOUSE": GOSUB
     11000: VTAB 5: RESTORE : FOR
     I = 1 TO 7: READ A$(I): PRINT
     :A$ = A$(I): GOSUB 11000: NEXT
     : PRINT : PRINT "  SELECT ON
     E OF THE ABOVE:";: GET A$: PRINT
     A$: IF A$ < "1" OR A$ > "7" THEN
      HOME : GOTO 200
230 A =  VAL (A$): ON A GOSUB 100
     0,2000,3000,4000,5000,6000,7
     000,: TEXT : HOME : CLEAR : GOTO
```

```
 110
1000  REM
1010  HOME : VTAB 12: PRINT " FIL
 E NAME :";: FOR I = 1 TO 10:
  PRINT  CHR$ (95);: NEXT : PRINT
 : VTAB 12: HTAB 13: INPUT ""
 ;A$: IF A$ = "" THEN  RETURN

1020  IF  LEN (A$) > 10 THEN 1000

1030  IF FL = 1 THEN FL = 0: GOTO
 1050
1040  GOSUB 15000
1050  FOR I = 1 TO CZ - 1: IF A$ =
 TI$(I) THEN 1080
1060  NEXT : HOME : VTAB 12: PRINT
 " THAT FILE DOES NOT EXIST."
 : PRINT : PRINT " TRY AGAIN
 ";: INPUT YN$: IF  LEFT$ (YN
 $,1) = "Y" THEN FL = 1: GOTO
 1010
1070  RETURN
1080  GOSUB 9000: POKE 35,21
1100  HOME : VTAB 22: INVERSE : PRINT
 "----------------------------
 -----";: VTAB 23: PRINT "
  
 ";: SPEED= 140: NORMAL : POKE
 232,3: FOR I = 1 TO MC - 1:A
 $ = ME$(I): INVERSE : VTAB 2
 3: PRINT "ITEM # "I;: HTAB 1
 6: PRINT "DATE :";: IF MD$(I
 ) = "" THEN  PRINT "        "
 ;: GOTO 1160
1140  IF  LEN (MD$(I)) < 8 THEN M
 D$(I) = " " + MD$(I): GOTO 1
 140
1150  HTAB 31 -  LEN (MD$(I)): PRINT
 MD$(I);
1160  NORMAL :LM = 2:RM = 32: FOR
 J = 0 TO 8: VTAB  PEEK (232)
  + J: HTAB 1: CALL  - 868: FOR
 L = 1 TO 100: NEXT : NEXT : VTAB
  PEEK (232): GOSUB 10000: IF
  PEEK (232) = 3 THEN  POKE 2
 32,12: GOTO 1220
1210  IF  PEEK (232) = 12 THEN  POKE
 232,3
1220  NEXT : FOR I = 1 TO 3000: NEXT
 : FOR I = 21 TO 2 STEP  - 1:
  VTAB I: CALL  - 868: FOR J =
 1 TO 100: NEXT : NEXT : GOTO
 1100
2000  REM
2010  HOME : VTAB 12: PRINT " FIL
 E NAME :";: FOR I = 1 TO 10:
  PRINT  CHR$ (95);: NEXT : PRINT
 : VTAB 12: HTAB 13: INPUT ""
 ;A$: IF A$ = "" THEN  RETURN

2020  IF  LEN (A$) > 10 THEN 2000

2030  IF FL = 1 THEN FL = 0: GOTO
 2050
2040  GOSUB 15000
2050  FOR I = 1 TO CZ - 1: IF A$ =
 TI$(I) THEN 2080
2060  NEXT : HOME : VTAB 12: PRINT
 " THAT FILE DOES NOT EXIST."
 : PRINT : PRINT " TRY AGAIN
 ";: INPUT YN$: IF  LEFT$ (YN
 $,1) = "Y" THEN FL = 1: GOTO
 2000
2070  RETURN
2080 AD = 1: HOME : GOSUB 3090
2090  PRINT : PRINT " IS THIS COR
 RECT ";: INPUT YN$: IF  LEFT$
 (YN$,1) = "N" THEN 2080
2100  PRINT  CHR$ (4)"APPEND "A$:
  PRINT  CHR$ (4)"WRITE "A$: PRINT
 ME$(MC): PRINT MD$(MC): PRINT
  CHR$ (4)"CLOSE": PRINT : PRINT
 " ANOTHER ";: INPUT YN$: IF
  LEFT$ (YN$,1) = "N" THEN AD
  = 0: RETURN
2120  HOME : GOSUB 3090: GOTO 209
 0
3000  REM
3010  GOSUB 15000: REM
3020  HOME
3030  VTAB 3: PRINT " FILE NAME :
 ";: FOR I = 1 TO 10: PRINT  CHR$
 (95);: NEXT : PRINT : VTAB 3
 : HTAB 13: INPUT "";TI$(CZ +
 1): IF  LEN (TI$(CZ + 1)) >
 10 THEN  HOME : GOTO 3030
3040  IF TI$(CZ + 1) = "" THEN 31
 70
3050 MC = 1: IF CZ = 0 THEN 3090
3070  FOR I = 1 TO CZ: IF TI$(CZ +
 1) = TI$(I) THEN  HOME : VTAB
 12:A$ = "THAT TITLE ALREADY
 EXIST.": GOSUB 11000: FOR I =
 1 TO 3000: NEXT : GOTO 3020
3080  NEXT
3090  INVERSE : VTAB 5: HTAB 2: PRINT
 "ENTER MESSAGE :": NORMAL : PRINT
 : FOR I = 1 TO 224: PRINT  CHR$
 (95);: NEXT : PRINT : VTAB 7
 : HTAB 1: GOSUB 8000:ME$(MC)
  = ST$: VTAB 15: PRINT " DAT
 E (MM/DD/YY) :";: FOR I = 1 TO
 8: PRINT  CHR$ (95);: NEXT :
  PRINT : VTAB 15: HTAB 19: INPUT
 "";MD$(MC): IF AD = 1 THEN  RETURN
3130  PRINT : INPUT " IS THIS COR
 RECT ?";YN$: IF  LEFT$ (YN$,
 1) = "N" THEN  HOME : GOTO 3
 090
3140  PRINT : PRINT " ANOTHER ?";
 : INPUT "";YN$: IF  LEFT$ (Y
 N$,1) = "Y" THEN MC = MC + 1
 : HOME : GOTO 3090
3150  PRINT  CHR$ (4)"OPEN "TI$(C
 Z + 1): PRINT  CHR$ (4)"WRIT
 E"TI$(CZ + 1): FOR I = 1 TO
 MC: PRINT ME$(I): PRINT MD$(
 I): NEXT : PRINT  CHR$ (4)"C
 LOSE": PRINT  CHR$ (4)"OPEN
 TITLES": PRINT  CHR$ (4)"POS
```

```
 ITION TITLES,R"CZ - 1: PRINT
  CHR$ (4)"WRITE TITLES": PRINT
 TI$(CZ + 1): PRINT "***": PRINT
  CHR$ (4)"CLOSE"
3170  RETURN
4000  REM
4010  HOME : VTAB 12: PRINT " FIL
 E NAME :";: FOR I = 1 TO 10:
  PRINT  CHR$ (95);: NEXT : PRINT
 : VTAB 12: HTAB 13: INPUT ""
 ;A$: IF A$ = "" THEN  RETURN
4020  IF  LEN (A$) > 10 THEN 4000
4030  IF FL = 1 THEN FL = 0: GOTO
 4060
4040  GOSUB 15000:FI$ = A$
4060  FOR I = 1 TO CZ - 1: IF A$ =
 TI$(I) THEN 4090
4070  NEXT : HOME : VTAB 12: PRINT
 " THAT FILE DOES NOT EXIST."
 : PRINT : PRINT " TRY AGAIN
 ";: INPUT YN$: IF  LEFT$ (YN
 $,1) = "Y" THEN FL = 1: GOTO
 4000
4080  RETURN
4090  GOSUB 9000: FOR I = 1 TO MC
  - 1:A$ = ME$(I)
4110  HOME : VTAB 3: PRINT " ITEM
 #";I: PRINT :LM = 2:RM = 32
 : GOSUB 10000: PRINT : PRINT
 " DATE :"MD$(I): VTAB 17: PRINT
 " CHANGE :": PRINT " (D)ATE
 (M)ESSAGE (P)URGE": PRINT "
 (N)EXT": VTAB  PEEK (37) - 2
 : HTAB 10: GET YN$: PRINT YN
 $: IF YN$ = "N" THEN 4180
4140 CH = 1: IF YN$ = "D" THEN 42
 60
4160  IF YN$ = "P" THEN 4330
4170  IF YN$ = "M" THEN 4290
4180  IF MC = 1 THEN 4210
4190  IF I > = MC - 1 THEN 4210
4200  NEXT
4210 A$ = FI$: IF CH = 0 THEN 424
 0
4220  PRINT  CHR$ (4)"DELETE "A$:
  PRINT  CHR$ (4)"OPEN "A$: PRINT
  CHR$ (4)"WRITE "A$: FOR I =
 1 TO MC - 1: PRINT ME$(I): PRINT
 MD$(I): NEXT : PRINT  CHR$ (
 4)"CLOSE"
4240  VTAB 15: CALL  - 958: PRINT
 " CHANGE ANOTHER";: INPUT YN
 $: IF  LEFT$ (YN$,1) = "Y" THEN
  HOME :CH = 0: GOTO 4000
4250  RETURN
4260  VTAB 15: CALL  - 958: PRINT
 " DATE (MM/DD/YY) :";: FOR J
  = 1 TO 8: PRINT  CHR$ (95);
 : NEXT : VTAB 15: HTAB 19: INPUT
 "";MD$(I): IF  LEN (MD$(I)) >
  8 THEN  VTAB 7: CALL  - 958:
  GOTO 4260
4280 A$ = ME$(I): GOTO 4110
4290  VTAB 13: CALL  - 958: PRINT
 " MESSAGE :": VTAB 15: FOR J
  = 1 TO 224: PRINT  CHR$ (95
 );: NEXT : VTAB 15: GOSUB 80
 00: IF  LEN (ST$) = 1 THEN A
 $ = ME$(I): GOTO 4320
4310 ME$(I) = ST$:A$ = ST$
4320  GOTO 4110
4330  VTAB 17: CALL  - 958: PRINT
 " ENTER 'DEL' TO CONFIRM :";
 : INPUT "";YN$: IF YN$ <  >
 "DEL" THEN 4110
4340  IF I = MC THEN MC = MC - 1:
  GOTO 4200
4350  FOR J = 1 TO MC - 1:ME$(J) =
 ME$(J + 1):MD$(J) = MD$(J +
 1): NEXT :MC = MC - 1:A$ = M
 E$(I): IF I > = MC THEN 418
 0
4400  GOTO 4110
5000  REM
5010  HOME : VTAB 12: PRINT " ENT
 ER FILE TO BE DELETED :": PRINT
 : HTAB 10: FOR I = 1 TO 10: PRINT
  CHR$ (95);: NEXT : PRINT : VTAB
  14: HTAB 10: INPUT "";A$: IF
 A$ = "" THEN  RETURN
5030  IF  LEN (A$) > 10 THEN 5010
5040  GOSUB 15000: FOR I = 1 TO C
 Z - 1: IF A$ = TI$(I) THEN 5
 090
5060  NEXT : HOME : VTAB 12: PRINT
 " UNABLE TO MATCH THAT NAME.
 ": PRINT : PRINT " TRY AGAIN
 ?";: INPUT "";YN$: IF  LEFT$
 (YN$,1) = "Y" THEN 5000
5080  RETURN
5090  PRINT : PRINT " ENTER 'DEL'
  TO DELETE FILE.": PRINT : HTAB
 10: INPUT YN$: IF YN$ <  > "
 DEL" THEN  RETURN
5100  HOME : VTAB 12: HTAB 5: INVERSE
 : PRINT "FILE BEING DELETED.
 ": NORMAL : IF I = CZ - 1 AND
 I = 1 THEN  PRINT  CHR$ (4)"
 DELETE TITLES": PRINT  CHR$
 (4)"DELETE "A$: RETURN
5120  FOR J = 1 TO CZ - 1:TI$(J) =
 TI$(J + 1): NEXT : PRINT  CHR$
 (4)"DELETE TITLES": PRINT  CHR$
 (4)"OPEN TITLES": PRINT  CHR$
 (4)"WRITE TITLES": FOR I = 1
 TO CZ - 1: PRINT TI$(I): NEXT
 : PRINT  CHR$ (4)"CLOSE": PRINT
  CHR$ (4)"DELETE "A$: PRINT
 : PRINT " FILE DELETED, DO A
 NOTHER ?";: INPUT "";YN$: IF
  LEFT$ (YN$,1) = "Y" THEN 50
 00
5160  RETURN
6000  REM
6010  GOSUB 15000:B = 1: HOME
6030  FOR I = B TO CZ - 1: IF I <
 10 THEN  HTAB 2
```

```
6050  IF  PEEK (37) < 22 THEN  PRINT
   I". "TI$(I): GOTO 6070
6060  GOTO 6090
6070  NEXT : GET A$: PRINT A$: RETURN
6090 J = I:I = CZ - 1: IF J = I THEN
   6150
6100  VTAB 2: FOR I = J TO CZ - 1
   : HTAB 16: IF  PEEK (37) < 2
   2 THEN  PRINT I". "TI$(I): GOTO
   6140
6130  GOTO 6150
6140  NEXT : HTAB 16: GET A$: PRINT
   A$: RETURN
6150  HTAB 16: GET A$: PRINT A$:B
   = I:I = CZ - 1: IF B = I THEN
   6140
6170  HOME : GOTO 6030
7000  TEXT : HOME : END
8000  REM
8010 ST$ = "": CALL  - 662: FOR J
   = 512 TO 736:X =  PEEK (J):
   IF X = 141 THEN 8060
8040 ST$ = ST$ +  CHR$ (X - 128):
   NEXT : IF  LEN (ST$) > 224 THEN
   PRINT : PRINT  CHR$ (7)" TO
   O LONG!": FOR J = 1 TO 3000:
   NEXT : HOME : POP : GOTO 30
   70
8060  IF ST$ = "" THEN  POP : HOME
   : GOTO 3070
8070  RETURN
9000  REM
9010 MC = 1: PRINT  CHR$ (4)"OPEN
   "A$: PRINT  CHR$ (4)"READ "
   A$
9030 ST$ = "": CALL  - 662: FOR I
   = 512 TO 736:X =  PEEK (I):
   IF X = 141 THEN 9090
9060  IF X > 141 THEN X = X - 128
9070 ST$ = ST$ +  CHR$ (X): NEXT
9090 ME$(MC) = ST$: INPUT MD$(MC)
   :MC = MC + 1: GOTO 9030
9110  PRINT  CHR$ (4)"CLOSE": RETURN
10000  REM
10010  IF  LEN (A$) <  = RM - LM THEN
   PRINT A$: RETURN
10020  FOR J = RM - LM TO 1 STEP
   - 1:C$ =  MID$ (A$,J,1): IF
   C$ = " " OR C$ = "." OR C$ =
   "," OR C$ = "!" OR C$ = "?" OR
   C$ = "-" THEN 10050
10040  NEXT :J = RM - LM
10050  IF C$ = " " THEN A1$ =  LEFT$
   (A$,J - 1):A$ =  RIGHT$ (A$,
   LEN (A$) - J): GOTO 10090
10060  IF C$ = "!" OR C$ = "," OR
   C$ = "." OR C$ = "?" OR C$ =
   "-" AND J = RM - LM THEN  GOSUB
   10130:A1$ =  LEFT$ (A$,J):A$
   =  RIGHT$ (A$, LEN (A$) - J
   - 1): HTAB LM: PRINT A1$: GOTO
   10100
10070 A1$ =  LEFT$ (A$,J): IF  LEN
   (A$) <  >  LEN (A1$) THEN A$
   =  RIGHT$ (A$, LEN (A$) - J
   ): GOTO 10090
10080  GOTO 10110
10090  HTAB LM: PRINT A1$
10100  IF  LEN (A$) > RM - LM THEN
   10020
10110  HTAB LM: PRINT A$: RETURN
10130  IF J =  LEN (A$) THEN  RETURN
10140  IF  MID$ (A$,J + 1,1) <  >
   " " THEN A1$ =  LEFT$ (A$,J)
   :A$ =  RIGHT$ (A$, LEN (A$) -
   J): HTAB LM: PRINT A1$;: POP
   : GOTO 10100
10150  RETURN
11000  PRINT  TAB( ((32) -  LEN (
   A$)) / 2)A$: RETURN
12000  DATA  "1. LOAD & DISPLAY F
   ILE","2. ADD TO A FILE
   ","3. ADD A NEW FILE     ","4.
   CHANGE A FILE       ","5. DEL
   ETE A FILE        ","6. VIEW N
   AME-FILE        ","7. EXIT PROG
   RAM        "
15000  REM
15010 CZ = 1: PRINT  CHR$ (4)"OPE
   N TITLES": PRINT  CHR$ (4)"R
   EAD TITLES"
15030  INPUT TI$(CZ): IF TI$(CZ) =
   "***" THEN 15050
15040 CZ = CZ + 1: GOTO 15030
15050  PRINT  CHR$ (4)"CLOSE"
15060  RETURN
32000  REM
32010 ER =  PEEK (222):EL =  PEEK
   (218) +  PEEK (219) * 256: CALL
   768: IF ER = 5 AND EL = 1503
   0 AND A <  > 3 THEN  PRINT :
   PRINT  CHR$ (4)"CLOSE": HOME
   : VTAB 12:A$ = "NO TITLES IN
   FILE.": GOSUB 11000: FOR I =
   1 TO 4000: NEXT : POP : RETURN
32040  IF ER = 5 AND EL = 15030 THEN
   PRINT : PRINT  CHR$ (4)"CLO
   SE":ER = 1: GOTO 15060
32050  IF ER = 5 AND EL = 9040 OR
   EL = 9100 THEN  PRINT : GOTO
   9110
32060  IF ER = 22 AND EL > 10000 THEN
   RESUME
32070  IF ER = 255 THEN  SPEED= 2
   55: GOTO 130
32080  HOME : IF ER = 9 THEN  PRINT
   "DISK FULL!": END
32100  IF ER = 10 THEN  PRINT "FI
   LE IS LOCKED!": END
32110  IF ER = 8 THEN  PRINT "DIS
   K I/O ERROR!": END
32120  IF ER = 4 THEN  PRINT "REM
   OVE WRITE PROTECT TAB!": END
```

讓九個月大的嬰兒接觸電腦吧！

自從微型電腦誕生於世後，在外國，就出現了不少的電腦神童，他們年紀輕輕，但對電腦操控的天份却可比美好多電腦工程師。今時今日，在香港，許多痛愛自己子女的父母，都抱著縱然自己不曉電腦，但下一代一定要懂得的心理。於是，許多年紀細細的兒童，在父母的鼓勵之下，對電腦頗具有心得。而許多父母，都會送自己的子女去參加兒童電腦班，情況有如要他們去學鋼琴或跳芭蕾舞般普遍。他們希望子女們能從小便接觸電腦，以便他日能勝人一籌。

到底，我們應該讓子女何時接觸電腦呢？就由他們九個月大開始吧！

不要以爲幾個月大的嬰兒甚麽也不曉，根據外國的兒童心理學家臨床研究，所有的嬰兒都是天生充滿好奇心，他們對四週環境的一切都會發生興趣，特別是對於一些強烈色彩而又會活動的東西。

雖然大部份的父母都會害怕小兒子們手多，弄壞精緻的電腦，有些還禁止他們觸碰電腦，直至他們表現得可信賴時才放手讓他們玩，但這時的小孩可能已經有七、八歲。

不過，在目前情況下，越早讓孩子接觸電腦似乎越好，一如嬰兒學游水一樣。

這個寫給九個月大至三歲年紀兒童的程序相信會令你的小兒產生非常的好奇，令他們會樂意接觸電腦。

每次小孩按下任何一個字符鍵或一SPACE BAR時，電腦都會清除低解像度彩色畫面，然後轉換一個色彩。就算是重覆按下該鍵，效果亦一樣，螢幕會先轉黑，然後出現色彩。

程式將每個字母鍵都指定有一個色彩。而ESC，RETURN ，及箭咀鍵會顯示黑色。

雖然，這個程式非常之簡單，但對於你及你的小孩都會帶來相當的樂趣。在你督導之下，你還可趁機教曉孩子們辨認簡單的英文單字及數字。

這個程式執行後是不會停止的，直至按CTRL RESET爲止。程式在程式磁碟上的FILENAME是BABY'S FIR—ST PROGRAM。

要注意，玩這遊戲必定要配備彩色電視機或監視器。■

```
430  REM  *********************
440  REM  **                 **
450  REM  **  BABY'S FIRST   **
460  REM  **  PROGRAM    BY  **
470  REM  **    COMPUHOUSE   **
480  REM  **                 **
490  REM  *********************
498  GET A$
499  GR
500  IF A$ = "A" THEN  COLOR= 1
501  IF A$ = "B" THEN  COLOR= 2
502  IF A$ = "D" THEN  COLOR= 3
503  IF A$ = "G" THEN  COLOR= 5
504  IF A$ = "F" THEN  COLOR= 4
505  IF A$ = "H" THEN  COLOR= 6
506  IF A$ = "J" THEN  COLOR= 7
507  IF A$ = "K" THEN  COLOR= 8
508  IF A$ = "L" THEN  COLOR= 9
509  IF A$ = ";" THEN  COLOR= 10
510  IF A$ = "X" THEN  COLOR= 11
511  IF A$ = "Z" THEN  COLOR= 12
512  IF A$ = "C" THEN  COLOR= 13
513  IF A$ = "V" THEN  COLOR= 14
514  IF A$ = "B" THEN  COLOR= 15
516  IF A$ = "N" THEN  COLOR= 1
517  IF A$ = "M" THEN  COLOR= 2
518  IF A$ = "M" THEN  COLOR= 3
519  IF A$ = "," THEN  COLOR= 4
520  IF A$ = "." THEN  COLOR= 5
521  IF A$ = "/" THEN  COLOR= 6
522  IF A$ = "Q" THEN  COLOR= 7
523  IF A$ = "W" THEN  COLOR= 7
524  IF A$ = "E" THEN  COLOR= 8
525  IF A$ = "R" THEN  COLOR= 9
526  IF A$ = "T" THEN  COLOR= 10
527  IF A$ = "Y" THEN  COLOR= 11
528  IF A$ = "U" THEN  COLOR= 12
529  IF A$ = "I" THEN  COLOR= 13
530  IF A$ = "O" THEN  COLOR= 14
531  IF A$ = "P" THEN  COLOR= 15
532  IF A$ = "P" THEN  COLOR= 15
533  IF A$ = "1" THEN  COLOR= 1
534  IF A$ = "2" THEN  COLOR= 2
535  IF A$ = "3" THEN  COLOR= 3
536  IF A$ = "4" THEN  COLOR= 4
537  IF A$ = "5" THEN  COLOR= 5
538  IF A$ = "6" THEN  COLOR= 6
539  IF A$ = "7" THEN  COLOR= 7
540  IF A$ = "8" THEN  COLOR= 8
541  IF A$ = "9" THEN  COLOR= 9
542  IF A$ = "0" THEN  COLOR= 10
543  IF A$ = ":" THEN  COLOR= 11
544  IF A$ = "-" THEN  COLOR= 12
545  IF A$ = " " THEN  COLOR= 1
1002 ZZ = 1
1006  HLIN 0,36 AT ZZ
1008 ZZ = ZZ + 1
1010  IF ZZ =  < 40 THEN  GOTO 10
     06
1011  PRINT "              "
1012  GOTO 498
```

符號逆輸型(RPN)計數機

你的ＡＰＰＬＥⅡ其實不單止用在大型程式上才能發揮它的優點，在計算功能上它亦十分優異，本程式是將Ａ—ＰＰＬＥ模擬作ＲＰＮ袋裝計算機。所謂ＲＰＮ是REVERSE POLISH NOTATION（符號逆輸入）的意思，它除了包括加、減、乘、除、指數，自然對數，開方根。這些基本運算外，還包括了堆疊器的運算，例如Ｘ與Ｙ的互相置換，將堆疊器向上捲升及捲下，監察暫存器的內容等。

ＲＰＮ計算機亦提供了錯誤追測系統及更改輸出格式變更的性能，而且大部份的功能都顯示在螢幕上，因此易於使用。

有關於ＲＰＮ的背景，據悉是十年前惠普HEWLETT PACKARD公司將其產品ＨＰ—35型袋裝函數計算機開始，取消了等於〝＝〞符號並以〝Ｅ—ＮＴＥＲ〞來取代，使數字在四則運算時不需要按入括號，而且運算符號可以跟隨在兩個數字之後按入。以下爲此系統的使用方法：

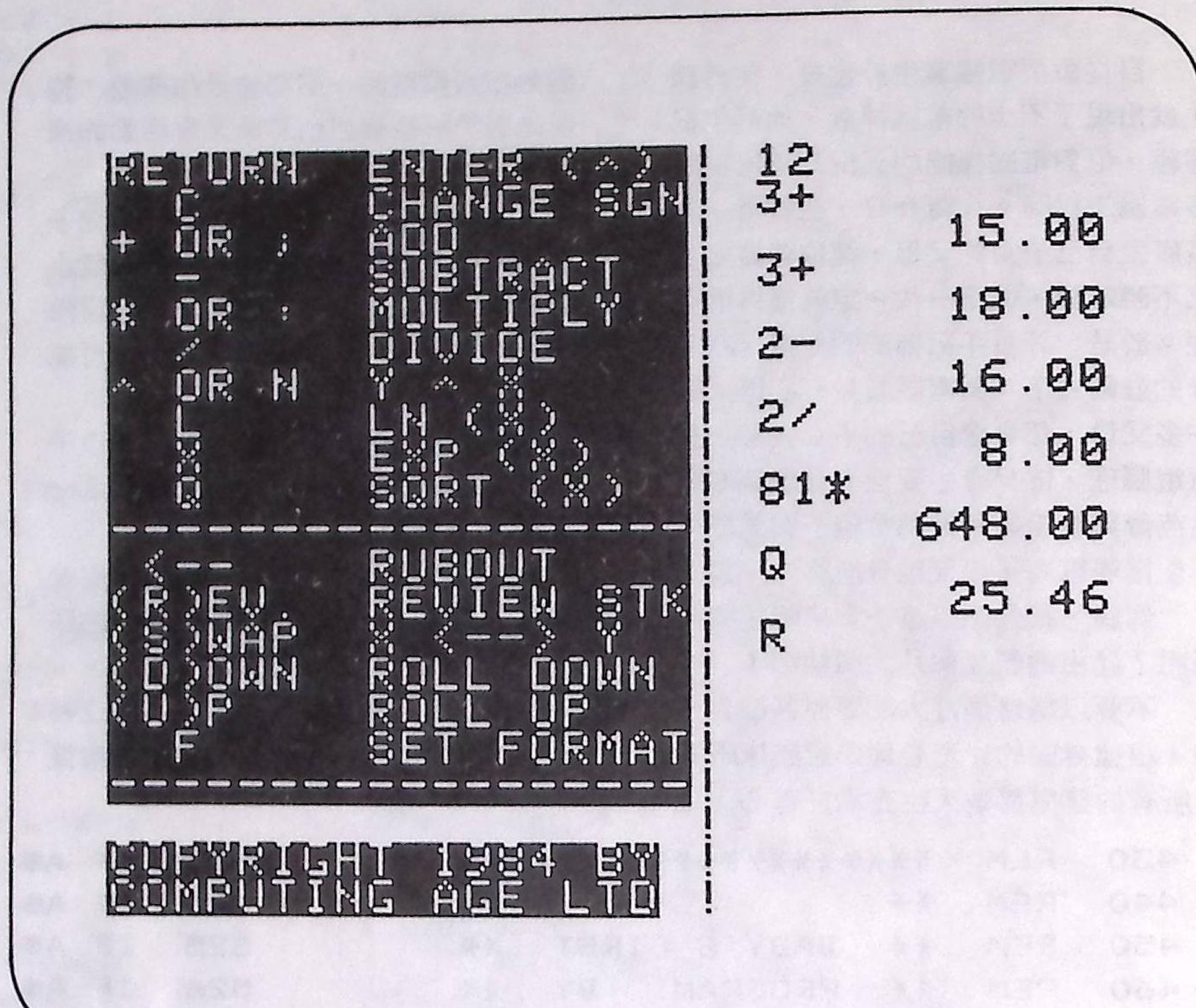
例子(1)　5＋10　的輸入方法是
　5〔RET〕
　10＋

例子(2)　(15＋3）×（40＋6）
　輸入方法是　15〔RET〕
　3＋
　40〔RET〕
　6＋×

ＲＰＮ計算機可作四則運算外，還有找出自然對數（NATURAL LOG)（按Ｌ），Ｅ×Ｐ（按Ｘ），及開方根（按Ｑ）與及多次開方（按＾或Ｎ）。附圖所見左邊爲各功能鍵表示圖，而右邊部份是一條數式計算例子。

ＲＰＮ計算機尚有其他優點，它能自動將輸出得數以千位或百萬位顯示。大家可從指令欄中看到，本計算機將數字輸出格式定爲小數後２個位。不過只要按ＦＦ，便可自行將格式重定（由０至９個位小數輸出）。

ＲＰＮ系統提供了四個額外的暫存器，它們分別是Ｘ、Ｙ、Ｚ，及Ｔ暫存器，它們會寄存最近四次的鍵入數字或輸出結果。



其它的命令如下：

Ｃ　可以將暫存器內或輸入數字改變正負號。

←　可以將錯誤按入而又未按ＲＥＴ—ＵＲＮ的數字更正。

Ｓ　將Ｘ及Ｙ暫存器的內容互相置換。

舉例說：

我們要計算10÷5

但錯手先按入５，然後10

此時，如果再按入１，則出來的答案是0.5，

我們可以先輸入Ｓ，然後按１，則可得出２

抱歉由於列表太長，這裏只刊登本程式的玩法，讀者可在本期程式磁碟中輕易找到程式的FILE。

檔案名字是Ｒ ＰＮ CALCULATOR

Ｄ　將暫存器的內容向上循環，如下圖

Ｔ
↑
Ｚ
↑
Ｙ
↑
Ｘ

Ｕ　與Ｄ的作用相反，將暫存器向下移，如下圖

Ｔ
↓
Ｚ
↓
Ｙ
↓
Ｘ

CTRL—E　囘到選擇程式目錄去。

徵求一行程式有獎遊戲

自〔徵求一行程式有獎遊戲〕刊出後，大家的反應非常踴躍。我們收到很多來信。

大部份的參加者都明白到遊戲的參加規則。在這些參加者中，水準不過不失，只有幾個較爲突出。許多參加者投寄的作品都是跟數學或物理學上的算式有關，我們覺得，這並不能表現出作者在BASIC 語言及監督程式的運用技巧。希望大家以後留意。由於篇幅所限，我們未能將落選的作品刊登出來，不過，所有這些落選的作品，我們將會全部陸續登在經已出版的COMPUSOFT磁碟月刊上。

落選的讀者請勿灰心，應繼續再嘗試參加。

今次，我們除刊登獲選作品外，還多舉出幾個例子程式，供大家觀摩參考的。■

歐陽偉誠（青衣島）可贏得一張第九期程式磁碟。

```
30  HGR2 : HCOLOR= 7: POKE 232,0:
    POKE 233,3: POKE 768,1: POKE
    769,0: POKE 770,4: POKE E771
    ,0: FOR L = 772 TO 778: POKE
    L, INT ( RND (1) * 255) + 1:
    NEXT L: POKE L,0: FOR L = 1
    TO 255: SCALE= INT (L / 10
    0) + 1: ROT= L: DRAW 1 AT 13
    9,95: NEXT L: PRINT  CHR$ (7
    ): GET A$: RUN
```

程式利用了APPLESOFT內的繪圖功能，先以亂數POKE入圖形表內，製成一個不規則的圖形後，再運用了APPLESOFT內兩個特別的指令——SCALE及ROT 使圖形依規律地變大及轉動，再將圖形畫在高解像度第2頁區中，構成一個美麗的圖案。假如改變圖形表的長度，SCALE值或ROT值，則可得到不同的效果。

（註：當聽到"嘟"聲後，再按鍵就可得到另一款圖案）

```
3  HGR : FOR A = .9 TO 2.9: HGR2
   : HCOLOR= 3: HPLOT 140,96: FOR
   B = 0 TO 3.1416 * 20 STEP .0
   5:R =  SIN (B * A): HPLOT  TO
   140 + 107 * R *  COS (B),96 +
   95 * R *  SIN (B): NEXT B: FOR
   C = 0 TO 99: POKE  - 16300,0
   : POKE  - 16299,0: NEXT : NEXT
```

本程式是以一個三角算式作爲基礎，加以變化，就能在高解像畫出三個精緻的幾何圖形，雖然，三個圖形的外形很相近，但是繪畫時的方式却完全不同，這就是最值得欣賞的地方了。

徐光榮（尖沙咀），請來電選擇那期程式磁碟。

WONG WAI LEUNG（沙田）可得到第12期程式磁碟一張。

本程式是一個跳字鐘，當RUN這程式時，鍵入時、分，對正時間便可按RETURN。跳字鐘相當準確。

```
3  INPUT "HOUR:";H: INPUT "MINUTE
   :";M: HOME : SPEED= 85: FOR
   I = 0 TO 1 STEP 0: FOR O = H
   TO 23: FOR N = M TO 59: FOR
   E = 0 TO 59: VTAB 16: PRINT
   O" ": VTAB 10: HTAB 18: PRINT
   ":"N" ": VTAB 10: HTAB 21: PRINT
   ":"E" ": NEXT E:M = 0: NEXT
   N:H = 0: NEXT O: NEXT I
```

DEMO1　這是一個可產生有趣的螢幕效果程式。

```
5  REM  ONE LINE PROGRAM DEMO 1
10  FOR I = 1 TO 1000: POKE  - 16
    299,0: HCOLOR= 3: POKE  - 16
    300,0: NEXT
```

DEMO2 此程式會有一個獨特的文字幕顯示 效果。

```
5  REM  ONE LINE PROGRAM DEMO 2
10  TEXT : HOME :A$ = "((()))": FOR
    X = 12 TO 23: HTAB X: VTAB X
    : PRINT A$: HTAB 24 - X: VTAB
    X: PRINT A$: HTAB X: VTAB 24
    - X: PRINT A$: HTAB 24 - X:
    VTAB 24 - X: PRINT A$: NEXT
    : FOR D = 1 TO 24: FOR Y = 1
    2 TO 23: VTAB D: HTAB Y: PRINT
    A$: VTAB D: HTAB 24 - Y: PRINT
    A$: NEXT : NEXT
```

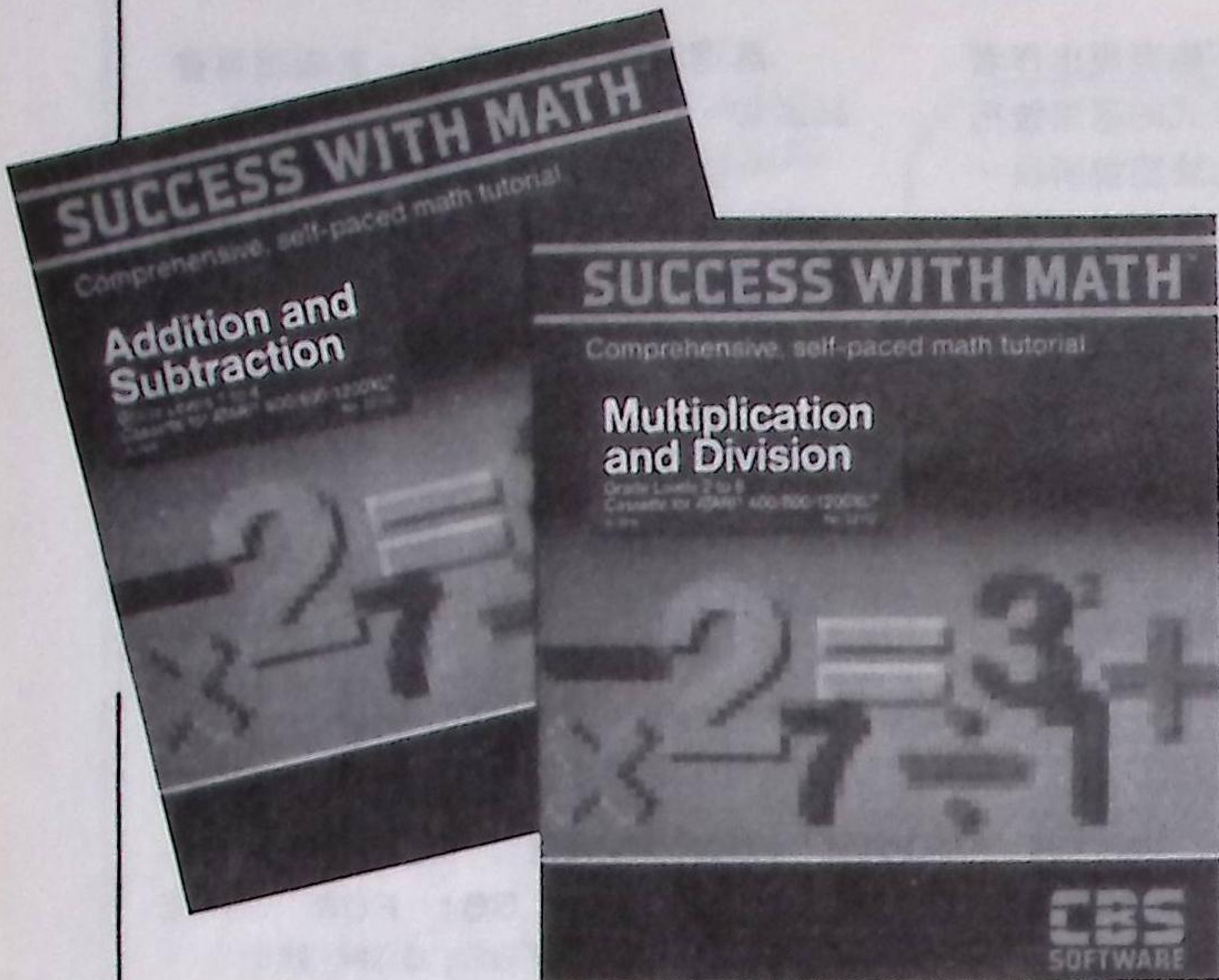
DEMO3　這程式能造出高解像動畫效果。

```
5  REM  ONE LINE DEMO 3
10  FOR A = 769 TO 783: READ B: POKE
    A,B: NEXT : HGR : HCOLOR= 5:
    HPLOT 0,0: CALL 62454: HGR2
    : CALL 769: DATA 173,87,192,
    234,173,84,192,234,173,85,19
    2,234,76,1,3
```

Software Reviews

教育眼

SUCCESS WITH MATH SERIES



軟件名稱：SUCCESS WITH MATH SERIES：
(1) MULTIPLICATION／DIVISION
(2) QUADRATIC EQUATION
出版公司：CBS SOFTWARE
適合機種：APPLE Ⅱ＋／ⅡE，IBM，COMMODORE 64 及 ATARI

電腦教育日益普及，港府目前正考慮於兩年後，在全港各中學施行電腦科，讓每名學生都有機會接觸電腦，了解電腦。

事實上，電腦不單是可利用來玩，透過電腦，可幫助我們學習其他知識，此知名爲電腦輔助教育。我們可利用電腦來學習化學、物理、英文、數學及其他學科，目前在外國有許多教育軟件。但是這些軟件是否適合本港教育環境則是一個疑問。

我們有見及此，開闢一欄專門評介目前已有或即將出版的教育軟件，以便讓師長、同學及關心子女的家長認識。

今期首先要介紹的是兩套自學數學軟件：

SUCCESS WITH MATH (MULTIPLICATION／DIVISION)及(QUADRATIC EQUATIONS)

俱爲CBS SOFTWARE 公司出品。前者適合小學一至六年級，後者適合中一至中三程度學生。

MULTIPLICATION／DIVISION 分爲兩部份：

① 1至3個位數字乘算
② 除數

在乘數學習部份中，被乘數是設定爲三個位，而乘數則可由用家自行選數1至3位數。

每回合可選擇最多九條題目。程式設計人在編寫時，設定了一個進位數，用家必須要自己計算進位，此舉可避免胡亂撞中答案。計算時，只容許一次錯誤，之後，電腦會自動將答案算出。每次結果算出後都會有附加評語。

至於第二部份的除算，用家可自行設定除數的數位（1至3個），被除數會視乎所除數的數位而定，它會較除數大兩個位。所有數都能被除盡。在計算時，編者所設步的運算步驟相當清楚。

所有過程都是文字字幕顯示，沒有利用高解像。

第二套要介紹的是QUADRATIC EQUATIONS，適合中一至中三程度。

程式是專門爲練習解QUADRATION EQUATION，有兩個程度。一般的算式是以$AX^2+BX+C=0$表達。

在較淺的程度中，系數A是1。

在較高的程度中，系數A會是2或3。

在解答問題時做法是要先將方程式化簡爲FACTOR，然後再計算X值。

筆者在試玩這軟件時，發覺作者在設計假若用家答案錯了後，電腦算出結果這部份相當好。對於並非太明白計算這類問題的同學，有一定的幫助。個人以爲，唯一美中不足之處，在於沒有一項用家自行輸入問題，而電腦幫助解答的功能。要編寫這部份事實上不算困難，爲何設計人不考慮。■

各期程式磁碟磁帶總目錄

第10期程式磁碟内容：

(1) 可隨意訂定格式之資料紀錄系統
(2) 太空戰士
(3) MARCO鍵盤
(4) 射鴨仔
(5) 奇妙的&——快速分類整理程式
(6) 40軌磁碟工具程式設計
(7) 變數表顯示
(8) 跳越加速器
(9) IQ方塊
(10) 快速檔案搜索
(11) 將DOS移上RAM咭
(12) 培養兒童的記憶能力
(13) 初中代數式運算
(14) 萬用數基轉換器
(15) 十六欄機械碼編印
(16) 捉蟲問題及答案

第11期程式磁碟内容：

(1) 家庭購物備忘及存貨紀錄系統
(2) 可隨意訂定之資料紀錄系統（下）
(3) 奇妙的&專欄——矩陣定值
(4) 40軌磁碟工具程式之(二) 單軌 INIT
(5) 兩個自動減省程式佔用之記憶空間應用程式
(6) 對我說
(7) 趣意盈盈的歡樂棒
(8) 英文語句文法分析練習系統
(9) 一行程式示例（共七個）
(10) 美化程式列表
(11) JACKIES HEAD
(12) 魔域迷宮
(13) 巨蟹求生記
(14) 自動日子記錄
(15) 空SEOCTOR圖表
(16) 捉虫訓練——FACTORIAL！

各期程式磁碟磁帶總目錄

1
1 突破傳統的鍵盤
2 捉球遊戲
3 勇闖電腦室
4 迷宮救美
5 將 APPLESOFT 變為 FORTRAN
6 上、下、左、右移動文字畫面
7 測試工具— BOOLEAN 程式
8 騰空更多磁碟空間一搬移HELLO
9 重現刪除之檔案
10 電腦時代月曆
11 COMPUTEXT 文字處理
12 時日相距計算程式
13 增強英文字滙能力
14 幼童數學四則運算練習
15 過四關

2
1 貨櫃處理
2 沙蟹幻方局
3 巧破密令
4 美術字繪劃器
5 小階字母處理
6 程式轉載器
7 彩色電視／監視器色彩測試
8 保留遊戲積分記錄
9 數學套件程式
10 磁碟修護工具
11 加送電腦時代月曆圖像一個

3
1 DOS 的自動同步追踪器
2 動量不滅定律模擬法
3 自製磁碟索引表封套
4 蘋果編曲器
5 幼童認數字初階
6 賽跑遊戲
7 步步追踪
8 DOS KILLER
9 圍棋遊戲
10 巴哈的古典樂章名作
11 旋轉螢幕畫面
12 雜誌目錄分類歸檔系統
13 神經病醫生一神經鬆弛體操

4
（1） DOS 操作指令簡化器
（2） TEXT FILE 有些甚麼內容？
（3） 立體迷宮
（4） 英文串字練習
（5） 如何以機械語言來修補 DOS
（6） 電腦動畫製作原理。
（7） 豐富設計的學前兒童教育程式
（8） 應用數學套件程式(二)
（9） DOS WRITER
（10） 穿梭機哥倫比亞號機
（11） 找回失落了的 STEP 及 TRACE
（12） TRACER控制常式
（13） 將 RAM 卡作為磁碟機使用
（14） 黑房計時器
（15） 太陽系模擬系統

9
（1）磁碟檔案黃頁系統。
（2）增強鍵盤輸入及螢幕輸出功能的系統
（3）將工具程式按放入 DOS 佔用的三條磁軌上
（4）快速尋找及磁碟矩陣儲存與提取
（5）新 DOS 命令改換程式
（6）任意設定目錄索引顯示模式
（7）捉迷藏
（8）古堡之謎
（9）高空擲物
（10）有趣的排序
（11）捉蟲問題及答案
（12）第八期捉蟲問題答案
（13）印出高解像數目字

5
1 私人約會備忘日曆
2 三度空間顯像系統
3 將 CP／M 檔案轉為 DOS
4 蘋果萬花筒
5 高解像度文字產生器
6 修訂 DOS 操作指令簡化器
7 濃縮圖片儲存空間
8 幼童數學練習—— 數字咭
9 加數機大登場
10 備忘 MEMO紙
11 光速計數器
12 剩餘磁碟空 SECTOR數目
13 16進制輸入數字鍵盤
14 問題棋盤
15 神偷智破夾萬
16 隔山打牛
17 大空船降落任務

6
（1）私人會計系統。
（2） APPLESOFT 擴展版本。
（3） GO]〔 HELLO 的程式
（4） APPLESOFT 偵錯系統。
（5） DOS 指令再修訂版本。
（6） 布倫時鐘。
（7） AMPERSAND 數基轉換器
（8） 完整索引錄。
（9） 砌積木。
（10） 神奇數字方塊。
（11） 亡命飛車。
（12） 迷宮搜秘。
（13） 太空怪侵略戰。
（14） 累加遊戲。
（15） 爲磁碟命名。
（16） 任意定義鍵盤。
（17） 捉蟲訓練問題。

7
（1） 會說話的蘋果
（2） 私人財政預算系統
（3） 私人收入會計系統
（4） 高解像字符繪劃器
（5） 圖像轉載器
（6） 螢幕文本編輯處理器
（7） 濃縮圖片程式修訂加強版本
（8） 蘋果杯大賽車
（9） 十六個難度的食鬼遊戲
（10） 高爾夫球遊戲。
（11） 空戰驚魂
（12） 美麗圖案設計家
（13） 更多色彩
（14） 列印的困擾
（15） 十一條空 SECTOR
（16） 捉蟲訓練
（17） 80字行咭例子程式

8
（一） APPLESOFT 滙編器
（二） 最簡單的拆帶程式
（三） 英文打字練習
（四） 編印單個LABEL
（五） 高解像頁處理工具箱
（六） 簡易快速的記憶移動方法
（七） 眞正要動腦筋的小學四則運算
（八） 七海霸王（DEPTH CHARGE）
（九） 原子黑洞
（十） 蛇之噩夢
（十一） 收銀機
（十二） 改良的閃動訊息
（十三） 兩欄索引顯示
（十四） 自動三角函數曲綫繪劃系統
（十五） 任意鍵盤（三個）
（十六） 每期捉蟲問題
（十七） 奇妙的 &

精選程式磁碟

編號：CA00A
內容：遊戲及教育程式
遊戲程式
1、捉球遊戲（第一期）
2、勇闖電腦室（第一期）
3、迷宮救美（第一期）
4、過四關（第一期）
5、沙蟹幻方局（第二期）
6、貨櫃處理（第二期）
7、賽跑遊戲（第三期）
8、步步追踪（第三期）
9、圍棋遊戲（第三期）
10、立體迷宮（第四期）
11、穿梭機哥倫比亞號（第四期）
教育程式
12、增強英文滙能力（第一期）
13、幼童數學四則運算（第一期）
14、數學套件程式（第二期）
15、動量不滅定律模擬法（第三期）
16、幼童認數字初階（第三期）
17、豐富設計的學前兒童教育程式（第四期
18、應用數學套件程式之二（第四期）

編號：CA00B
內容：應用程式精選
1、電腦時代月曆（第一期）
2、COMPUTEXT 文字處理（第一期）
3、美術學繪劃器（第二期）
4、彩色電視／監視器色彩測試（第二期）
5、保留遊戲積分紀錄（第三期）
6、自製磁碟索引封套（第三期）
7、APPLE 編曲器（第三期）
8、雜誌目錄分類歸檔系統（第三期）
9、DOS WRITER（第四期）
10、電腦動畫製作原理（第四期）
11、將 RAM 咭作為磁碟機使用（第四期）

各期程式磁帶

編號：C#1
1、捉球遊戲
2、勇闖電腦室
3、迷宮救美
4、過四關
5、增強英文字滙能力
6、幼童數學四則運算
7、時日相距計算程式

編號：C#2
1、沙蟹幻方局
2、貨櫃處理
3、數學套件程式
4、巧破密令
5、彩色電視／監視器色彩測試

編號：C#3
1、賽跑遊戲
2、步步追踪
3、圍棋遊戲
4、動量不變定律模擬方法
5、幼童認數字初階

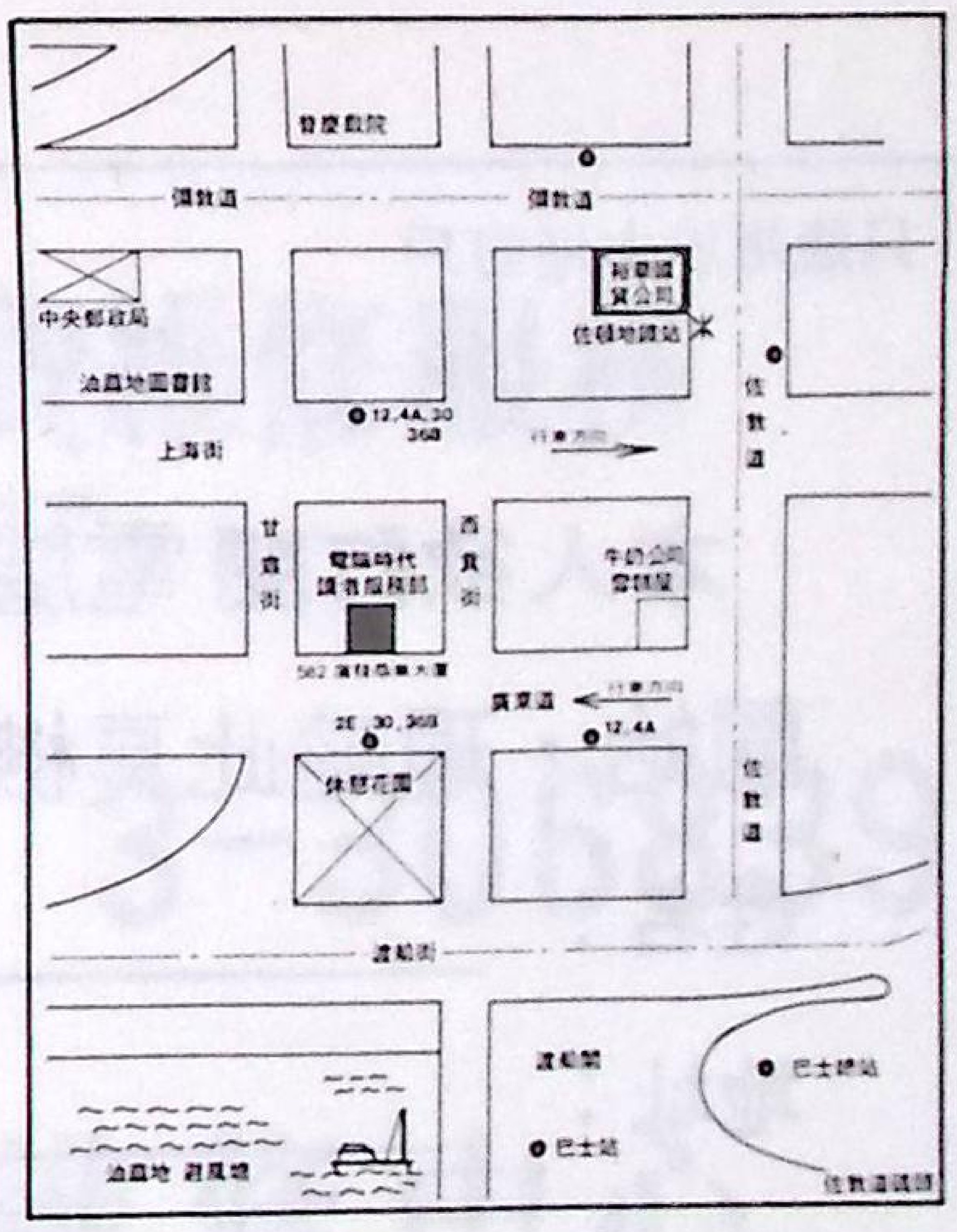
編號：C#4
1、豐富設計的學前兒童教育程式
2、應用數學套件程式之二
3、立體迷宮
4、穿梭機哥倫比亞號
5、英文串字練習

編號：C#5
1、加數機大登場
2、神偷智破夾萬
3、隔山打牛
4、太空船降落任務
5、蘋果萬花筒
6、幼童數學練習 ——數字咭

編號：C#6
1、累加遊戲
2、亡命飛車
3、迷宮搜秘
4 太空侵略戰
5、切積木
6、神奇數學方塊

編號：C#7
（一） 十六種難度的食鬼遊戲
（二） 更多色彩
（三） 空戰驚魂
（四） 捉虫問題
（五） 捉虫答案

編號：C#8
（一） 最簡單的拆帶程式
（二） 英文打字練習
（三） 原子黑洞
（四） 蛇之噩夢
（五） 收銀機
（六） 任意鍵盤



原價H.K.$455.
特價H.K.$365.
節省20%

現特優待長期讀者
立即訂閱可節省20%

《電腦時代》一本蘋果電腦用家必備雜誌

每期內容包括：

- 多個價值無可估計的軟件程式
- 專文深入討論週邊硬件的用途
- 最新的軟件介紹及評論
- 多個有深度而題材廣泛的專欄
- 還有其他精彩內容未能在此一一盡錄

每個程式都是精彩，每篇文章均有價值

*把握機會，請即填妥以下之《訂閱優待證》，寄九龍中央郵箱71193號，支票收款人請填寫"Computing Age Publisher"信封面請注明「訂閱」二字。

（請用英文填寫，方便電腦處理）

只適用於本港訂戶

•訂閱優待證•

本人欲訂閱 電腦時代雜誌及程式磁碟共十二期，由第____期開始，現趁此良機，本人只需付出H.K.$365.

姓名： ______________________

地址： ______________________________________

電話： ____________________

附上劃線支票No. ____________________

（如是舊訂戶續訂，請寫上訂閱號碼）**No.** ____________

本訂閱優待證只適用於雜誌連磁碟的訂戶，在十二月卅一日前有效。

何必自己動手

免費 由專業人員上門爲你修理微型電腦



請即加入本公司之保養及免費維修服務

24小時熱綫電話服務：3-306839

微型電腦保養維修中心

ROM & RAM computer ltd

SHOP: 582–592, Nathan Road, Basement Shop No. B35 Sino Centre, Kowloon. Tel: 3-306839

OFFICE: Rm. 1905 Witty Coml. Bldg., 1A–1J, Tung Choi St., Kowloon. TEL: 3-848129

Telex: 57551 ROMHX